



Analyses des motifs accessibles dans les modèles Kappa

Jérôme Feret

► To cite this version:

Jérôme Feret. Analyses des motifs accessibles dans les modèles Kappa. Cédric Lhoussaine et Élisabeth Remy. Approches symboliques de la modélisation et de l'analyse des systèmes biologiques, ISTE, 2022, 9781789480290. hal-03088539

HAL Id: hal-03088539

<https://inria.hal.science/hal-03088539>

Submitted on 2 Feb 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Analyses des motifs accessibles dans les modèles Kappa

Jérôme FERET^{1,2}

¹Département d'informatique de l'ÉNS, ÉNS, CNRS, Université
PSL, Paris, France

²INRIA, Paris, France

May 2020

Contents

1	Introduction	2
1.1	Contexte et motivations	2
1.2	Les langages de modélisation de systèmes d'interactions moléculaires	2
1.3	Le langage Kappa	4
1.4	Interprétation abstraite	7
1.5	L'écosystème Kappa	9
1.5.1	Analyses statiques	9
1.5.2	Analyses causales	10
1.5.3	Réduction de modèles	14
1.6	Contenu du chapitre	16
2	Graphes à sites	16
2.1	Signature	16
2.2	Complexes biochimiques	18
2.3	Motifs	19
2.4	Plongements entre motifs	20
3	Réécriture de graphes à sites	21
3.1	Règles d'interaction	22
3.2	Réactions induites par une règle d'interaction	23
3.3	Réseaux de réactions sous-jacents	24
4	Analyse des motifs accessibles	25
4.1	Accessibilité dans un réseau réactionnel	26
4.2	Abstraction d'un ensemble d'états	27
4.3	Transferts de point-fixes	31

5	Analyse par ensembles de motifs orthogonaux	33
5.1	Ensembles de motifs orthogonaux	33
5.2	Post-traitement et visualisation des résultats	35
5.3	Étude de performance et utilisation concrète	36
6	Conclusion	38
7	Bibliography	40

1 Introduction

1.1 Contexte et motivations

Décrire et analyser les systèmes à grande échelle et fortement combinatoires qui sont issus de certains modèles mécanistiques de biologie des systèmes est encore hors de portée de l'état de l'art. Dans de tels modèles, le comportement individuel des occurrences de protéines, qui peuvent établir des liaisons et modifier leur capacité d'interaction, est influencé par des compétitions pour des ressources communes. De plus, les occurrences de protéines peuvent former une grande diversité de complexes biochimiques différents. La concurrence entre des interactions à différentes échelles de temps génère des boucles de rétro-actions non linéaires qui contrôlent l'abondance de ces complexes biochimiques. Enfin, ces systèmes font intervenir des interactions entre de très petites molécules, comme des ions ou des ligands et des complexes biochimiques gigantesques comme les brins d'acide désoxyribonucléique, le ribosome, ou le signalosome. Comprendre comment le comportement collectif des populations de protéines qui définit le phénotype, est engendré par le comportement individuel des occurrences de ces protéines reste un problème largement ouvert et un enjeu crucial.

Alors que les progrès technologiques permettent d'obtenir rapidement une quantité toujours plus importante de détails à propos des interactions mécanistiques potentielles entre les occurrences de protéines, et ce, à un prix très accessible, la communauté scientifique est encore bien loin de comprendre globalement comment le comportement macroscopique des systèmes dans leur ensemble émerge de ces interactions. C'est l'objectif annoncé de la biologie des systèmes. Mais ce but est sans espoir à moins que des méthodes spécifiques et innovantes pour décrire ces systèmes complexes et analyser leur propriété ne soient conçues. Bien entendu, ces méthodes devront passer à l'échelle de la très grande quantité d'informations qui est publiée dans la littérature à un rythme qui augmente de manière exponentielle.

1.2 Les langages de modélisation de systèmes d'interactions moléculaires

Les langages formels ont été beaucoup utilisés pour décrire des modèles d'interactions mécanistiques entre occurrences de protéines. Ils procurent des outils mathématiques pour traduire ces interactions et définir rigoureusement le comportement des systèmes ainsi représentés grâce à un choix de sémantiques qualitatives, stochastiques ou différentielles.

Les langages tels que les réseaux réactionnels (Feinberg) ou les réseaux de Petri classiques (Heiner et Koch 2004), se basent sur le paradigme de la réécriture multi-ensemble. Les interactions consistent à consommer des réactifs en échange de produits. Des constantes cinétiques permettent de préciser soit la vitesse, soit la fréquence moyenne – selon le choix de la sémantique – d'application des différentes réactions. Ceci les rend très utiles pour décrire et formaliser le comportement de systèmes d'interactions de petite ou moyenne taille. Cependant, ces langages peinent à représenter de grands modèles car ils ont besoin d'un nom (ou d'un emplacement dans le cas des réseaux de Petri) par type de complexes biomoléculaires.

Des langages de plus haut niveau, inspirés des différents paradigmes de programmation, tels que les tableaux d'états à messages (Damm et Harel 2001), les automates communicants (Plateau 1985), les algèbres de processus (Regev et collab. 2001 ; Ciocchetta et Hillston 2009), les langages orientés objet (Dematté et collab. 2008), les réseaux de Petri colorés (Gao et collab. 2011) et la réécriture de graphes à sites (Danos et Laneve 2004 ; Faeder et collab. 2005 ; Andrei et Kirchner 2008 ; John et collab. 2011), exploitent le fait que les interactions dépendent généralement de conditions locales sur les configurations des occurrences de protéines au sein des complexes biochimiques. Ces langages permettent ainsi de traduire les systèmes d'interactions entre les occurrences de protéines de manière plus parcimonieuse : seuls les détails qui importent pour une interaction donnée sont mentionnés pour décrire cette interaction.

Il est important de distinguer les approches basées sur les agents de celles basées sur les règles de réécriture. Dans les approches basées sur les agents, chaque entité, que ce soit un processus (Ciocchetta et Hillston 2009) ou un objet (Dematté et collab. 2008), doit contenir la description de tous ses comportements possibles. Les changements entre les configurations des différentes entités se synchronisent par le biais de règles de communication. Ces règles, généralement en très petit nombre, définissent la sémantique opérationnelle des langages. Il est possible de conditionner le comportement d'un agent à des propriétés de l'état d'un autre agent auquel cet agent serait lié, mais cela nécessite de recourir à des processus fictifs pour aller chercher cette information. Cette astuce était en fait déjà utilisée dans les premiers modèles décrits en π -calcul (Regev et collab. 2001). Cependant, en général, les approches basées sur les agents donnent lieu à des systèmes de processus à états finis (Kahramanogullari et Cardelli 2013). Ceci permet d'étudier leur comportement à l'aide d'outils de vérification symbolique de modèles comme PRISM (Kwiatkowska et collab. 2011).

Lorsque les occurrences des protéines admettent trop de configurations différentes ou lorsque leurs capacités d'interaction dépendent trop des occurrences des protéines auxquelles elles sont liées, les approches fondées sur les agents ne passent pas à l'échelle, tant au niveau de la description des modèles que pour le calcul de leurs propriétés.

Dans les approches fondées sur les règles, les modèles sont définis par des règles d'interaction. Chaque règle définit sous quelles conditions sur les configurations des agents une interaction peut avoir lieu et quels sont les effets de cette interaction. Ainsi l'état des agents ne définit pas une fois pour toute les capacités d'interaction de cet agent. Ce sont les règles du modèle qui le font. Il n'est pas non plus nécessaire de donner la liste exhaustive de toutes les configurations des agents. Les règles peuvent se contenter de ne mentionner que les parties importantes des agents pour l'interaction qu'elles décrivent. Les approches fondées sur les règles passent mieux à l'échelle et facilitent la mise à jour des modèles. De plus, comme il n'est pas nécessaire de spécifier explicitement toutes les capacités d'interaction des occurrences des protéines, elles encouragent à une modélisation sans *a priori* où les interactions émergent des règles lors de la conception du modèle.

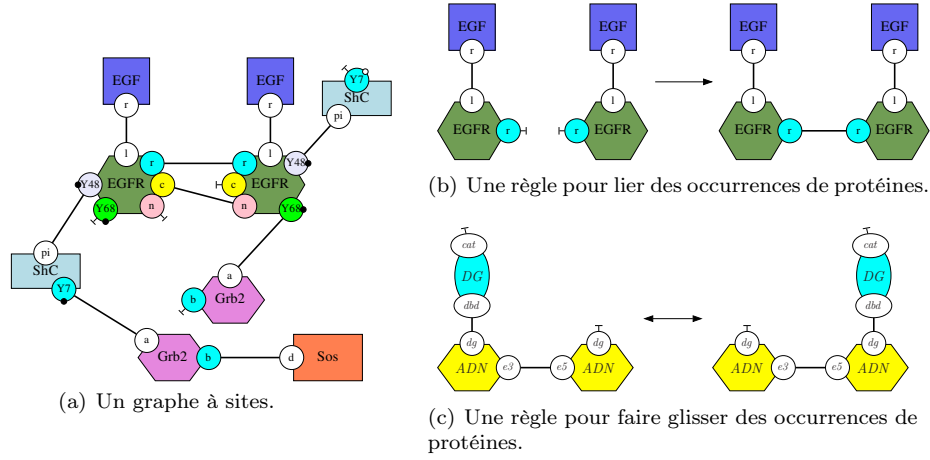


Figure 1: En 1(a) est dessiné un graphe à site. Il s'agit d'un complexe biochimique composé de deux occurrences du ligand (*EGF*), de deux occurrences du récepteur membranaire (*EGFR*), d'une occurrence de la protéine d'échafaudage (*Shc*), de deux occurrences de la protéine de transport (*Grb2*) et d'une occurrence de la protéine *Sos*. En 1(b) est donné un exemple de règle de liaison. Deux occurrences du récepteurs membranaires (*EGFR*), lorsqu'ils sont tous deux activés par une liaison avec des occurrences du ligand (*EGF*), peuvent se lier. Les autres sites sont omis car ils ne jouent aucun rôle dans cette interaction. En 1(c) est donnée une règle de déplacement. Une occurrence de l'enzyme Glycolase (*DG*) peut glisser dans les deux directions (selon une marche aléatoire) le long d'un brin d'ADN.

Le calcul des ambients (Cardelli et Gordon 1998, 2000), des bioambients (Regev et collab. 2004) et celui des membranes (Cardelli 2004) sont un peu particuliers. Ils permettent de décrire des boîtes ou des compartiments, qui peuvent être arbitrairement imbriqués au sein d'une arborescence, alors que des agents, contenus dans les boîtes dans le cas des ambients, ou dans leurs parois dans le cas des membranes, permettent à ces compartiments de se déplacer ou de se fusionner. Les capacités d'interaction des agents peuvent alors dépendre de leur localisation dans la hiérarchie des compartiments. La calcul projectif des membranes (Danos et Pradalier 2004) représente plus fidèlement la disposition des compartiments au sein d'une cellule, en rendant la description de l'état du système indépendante du choix de la racine de l'arborescence des compartiments.

1.3 Le langage Kappa

Les langages de réécriture de graphes à sites (Danos et Laneve 2004 ; Faeder et collab. 2005 ; Andrei et Kirchner 2008 ; John et collab. 2011) permettent de représenter de manière transparente les réseaux d'interactions entre des occur-

rences de protéines grâce à leur syntaxe qui est fortement inspirée de la chimie.

Dans Kappa, chaque complexe biochimique est représenté par un graphe à sites. Un exemple de graphe à sites est donné en Fig. 1(a). Dans un graphe à sites, des nœuds qui représentent des occurrences de protéines, sont associés à une liste de sites d'interaction. Ces sites peuvent être libres ou liés deux à deux. En outre, certains sites portent une propriété, qui peut servir à représenter un niveau d'activation. Les interactions entre occurrences de protéines peuvent modifier leurs conformations en dépliant ou en repliant leurs chaînes de nucléotides, ce qui peut révéler ou cacher des sites d'interaction. Dans Kappa, la structure tri-dimensionnelle des occurrences de protéines n'est pas représentée explicitement. En revanche, les conditions pour qu'un site d'interaction soit visible sont spécifiées dans la description des interactions elles-mêmes.

L'évolution d'un système Kappa se décrit grâce à des règles de réécriture hors-contexte. En Fig. 1(b) est dessinée une règle pour la formation de dimers. Deux récepteurs (*EGFR*) qui sont tous deux liés à des ligands (*EGF*) peuvent se lier entre eux pour former un dimer. En Fig. 1(c) est donnée une autre règle issue d'un modèle de réparation de l'ADN, dans laquelle une enzyme, la Glycolase (*DG*), peut glisser aléatoirement dans les deux sens, le long d'un brin d'ADN (Köhler et collab. 2014).

Une règle peut être comprise de manière intentionnelle comme une transformation locale de l'état du système ou de manière extensionnelle comme l'ensemble, qu'il soit fini ou non, des réactions biochimiques qui peuvent être obtenues en spécifiant entièrement les différents contextes d'application de ces règles. De cet ensemble de réactions, diverses sémantiques peuvent être définies pour décrire le comportement des systèmes. Ces sémantiques peuvent être qualitatives, stochastiques ou différentielles comme pour le cas des réseaux réactionnels et des réseaux de Pétri (les sémantiques quantitatives — stochastiques ou différentielles — nécessitent d'associer une constante de temps à chaque règle). Il est toutefois possible de simuler un modèle Kappa directement, sans passer par le réseau réactionnel sous-jacent. La simulation consiste alors à itérer la boucle événementielle suivante (celle-ci correspond à l'algorithme de Gillespie (Gillespie 1977)). Étant donné l'état du système, représenté par un graphe à sites, l'ensemble de tous les événements possibles est calculé. Un événement consiste à appliquer une règle dans le graphe à une occurrence du motif qui constitue le membre gauche de cette règle. Chaque événement a une propension qui correspond à la constante de la règle correspondante. Le prochain événement est tiré au hasard selon une probabilité proportionnelle à sa propension, alors que le délai entre deux événements est tiré aléatoirement selon une loi exponentielle dont le paramètre est la somme des propensions de tous les événements potentiels du système. Il n'est pas raisonnable de recalculer la liste des événements potentiels à chaque fois après l'application d'une règle. Cet ensemble peut être mis à jour dynamiquement en tenant compte uniquement des nouveaux événements potentiels et des événements qui ne sont plus possibles du fait de l'application du dernier événement choisi (Danos et collab. 2007b). Le simulateur actuel tire profit au maximum des sous-motifs communs dans les motifs qui apparaissent dans le membre gauche des règles pour découvrir les

nouveaux événements et retirer les événements devenus obsolètes plus rapidement (Boutillier et collab. 2017).

Le langage Kappa souffre de plusieurs limites. Par exemple, dans Kappa, les sites d'interaction d'une même occurrence d'une protéine doivent porter des noms différents ; par ailleurs, en ce qui concerne les propriétés géométriques, Kappa ne permet ni de représenter la structure tridimensionnelle des occurrences de protéines, ni leur répartition dans l'espace. Avoir des sites deux à deux différents dans chaque occurrence de protéines facilite grandement la recherche des occurrences des motifs dans les graphes, ce qui est non seulement crucial pour simuler les modèles de manière efficace, mais est aussi à la base de plusieurs constructions utilisées pour l'analyse statique et la réduction de modèles. Certains langages lèvent cette contrainte soit directement comme dans les langages BNGL (Faeder et collab. 2005) et *mød* (Andersen et collab. 2016), soit indirectement en utilisant un codage sous forme d'hyperliens comme c'est possible dans le langage React(C) (John et collab. 2011). Toutefois, l'efficacité des moteurs de simulation est fortement réduite quand de telles constructions sont utilisées. Pour ce qui est de la géométrie des protéines, les conditions liées aux conformations spatiales des protéines peuvent être encodées dans les règles de réécriture. Certaines extensions du langage permettent de représenter des contraintes sur la position relative des occurrences de protéines et des sites d'interaction dans les complexes biochimiques afin de restreindre l'ensemble des événements possibles à ceux qui satisfont ces contraintes (Danos et collab. 2015). Enfin, dans Kappa, la distribution des occurrences de protéines dans l'espace est passée sous silence. Il est fait l'hypothèse que les occurrences de protéines sont parfaitement mélangées. Il est donc impossible de retrouver les phénomènes d'encombrement qui peuvent être dus à des accumulations d'occurrences de protéines dans certaines régions de la cellule. De même, les gradients de concentration locaux qui pourrait être dus à la présence d'une occurrence d'une protéine d'échafaudage ne peuvent pas être représentés (en Kappa, chaque occurrence d'une protéine d'échafaudage n'agit qu'en maintenant des occurrences de protéines dans le même complexe biochimique, une fois libérée, ces occurrences de protéines ne sont pas supposées rester, même pour un court instant dans le même voisinage). Une solution partielle consiste à encoder en Kappa une grille pour représenter de manière discrète les positions potentielles des occurrences de protéines. Ensuite, celles-ci peuvent glisser le long de cette grille grâce à des règles implémentant la diffusion des occurrences de protéines. Le langage SpatialKappa (Stewart 2010) permet d'utiliser ce procédé de manière transparente. Par ailleurs, le langage ML (Helms et collab. 2017) permet de représenter des modèles d'interactions entre occurrences de protéines qui peuvent se déplacer de manière continue dans un milieu. Il est possible de munir un modèle Kappa d'un ensemble de compartiments statiques. Toutefois, ceci ne permet pas de modéliser le transport d'occurrences de protéines par le biais de vésicules. La machine formelle cellulaire (Damgaard et collab. 2012) répond à cet enjeu, sans toutefois fournir de moteurs de simulation efficaces.

Les langages de réécriture de graphes à sites permettent de représenter les réseaux d'interactions entre occurrences de protéines, et ce, malgré leur forte

combinatoire. Si le comportement de ces réseaux peut être formellement défini et simulé, des abstractions sont toutefois nécessaires pour calculer les propriétés du comportement collectif des populations de protéines.

1.4 Interprétation abstraite

L'interprétation abstraite a été introduite il y a maintenant un peu plus de quarante ans comme un cadre mathématique pour établir des liens formels entre le comportement de programmes, vu à différents niveaux d'abstraction. Depuis, l'interprétation abstraite a été utilisée non seulement pour comparer différentes méthodes et algorithmes d'analyse statique (Cousot 2002), mais aussi pour développer des analyseurs statiques qui peuvent calculer automatiquement les propriétés sur le comportement des programmes (Blanchet et collab. 2003 ; Fähndrich et Logozzo 2010). L'interprétation abstraite s'est désormais développée dans l'industrie (entre autres, Amazon, Facebook, IBM, Google, Microsoft et MathWorks ont chacune leurs propres analyseurs statiques basés sur l'interprétation abstraite).

L'interprétation abstraite repose sur la démarche suivante. Le comportement d'un programme (ou d'un modèle) peut en général être décrit comme le plus petit point fixe $\text{lfp } \mathbb{F}$ d'un opérateur \mathbb{F} agissant sur les éléments d'un ensemble appelé le domaine concret D . Le domaine concret est habituellement l'ensemble des parties $\wp(S)$ d'un ensemble d'éléments S , qui peuvent être des états, des traces de calcul, *et cetera*. Une abstraction est alors vue comme un changement de granularité dans la description du comportement des programmes (ou des modèles) et ce changement de granularité peut être représenté en langage mathématique sous diverses formes telles qu'un opérateur de clôture supérieure, une famille d'idéaux, une famille de Moore ou une correspondance de Galois. Les correspondances de Galois se sont vite imposées comme l'outil le plus populaire pour décrire une interprétation abstraite. Un changement du niveau d'observation du comportement d'un programme (ou d'un modèle) peut ainsi être décrit en choisissant un ensemble D^\sharp de propriétés d'intérêt. C'est le domaine abstrait. Cet ensemble est ordonné par un ordre partiel \sqsubseteq . Chaque élément a^\sharp de ce domaine abstrait représente intentionnellement l'ensemble des éléments concrets qui satisfont cette propriété. Cet ensemble est noté $\gamma(a^\sharp)$. La fonction γ , ainsi définie, est croissante (si $a^\sharp \sqsubseteq b^\sharp$, alors $\gamma(a^\sharp) \subseteq \gamma(b^\sharp)$). Ainsi, l'ordre \sqsubseteq représente le niveau d'information.

Un élément abstrait a^\sharp est dit être une abstraction d'un ensemble a d'éléments concrets, si et seulement si a est un sous-ensemble de l'ensemble $\gamma(a^\sharp)$. Une correspondance de Galois est obtenue quand chaque sous-ensemble a de l'ensemble S admet une meilleure abstraction, c'est à dire, que pour chaque partie a de l'ensemble S , il existe un élément abstrait, noté $\alpha(a)$ qui est d'une part une abstraction de l'ensemble a et d'autre part, qui est plus petit (pour l'ordre \sqsubseteq) que n'importe quelle abstraction de l'ensemble a . Dans un tel cas, n'importe quelle fonction croissante \mathbb{F}^\sharp opérant sur le domaine abstrait D^\sharp et telle que $[\alpha \circ \mathbb{F} \circ \gamma](a^\sharp) \sqsubseteq \mathbb{F}^\sharp(a^\sharp)$ pour chaque élément abstrait $a^\sharp \in D^\sharp$, admet un plus petit point fixe (pour l'ordre \sqsubseteq) noté $\text{lfp } \mathbb{F}^\sharp$. De plus, la concrétisation de ce

plus petit point fixe est un sur-ensemble du plus petit point fixe de la fonction \mathbb{F} ; ainsi le comportement du programme ou du modèle peut être calculé dans le domaine abstrait au prix d’une perte potentielle d’information puisque le résultat final est un sur-ensemble de l’ensemble de tous les comportements possibles. Par construction, l’approche est correcte : aucun comportement de la sémantique concrète n’est oublié. Par contre, quand le sur-ensemble ainsi calculé est un sur-ensemble strict, des comportements fictifs ont été introduits par l’analyse.

Le choix du domaine abstrait est crucial. Du point de vue de l’expressivité, le domaine abstrait doit permettre de décrire les propriétés d’intérêt des programmes (ou des modèles) ainsi que les propriétés intermédiaires qui sont nécessaires pour en établir la preuve de manière inductive. D’un point de vue algorithmique, ils doivent correspondre à des propriétés qui sont relativement simples à manipuler en machine. Enfin, la structure des chaînes croissantes d’éléments abstraits (pour l’ordre \sqsubseteq) est également importante pour que puissent être définis des opérateurs d’extrapolation précis, dans le cas où le domaine admettrait des chaînes croissantes infinies.

Plusieurs interprétations abstraites ont été proposées pour calculer automatiquement les propriétés des modèles en biologie des systèmes. Les premières ont naturellement été inspirées par les analyses de flot d’information (Bodei et collab. 1998 ; Feret 2000) et de dénombrement (Nielson et Nielson 2000 ; Feret 2001) dans le π -calcul et le calcul des ambients. Ces analyses permettent de détecter avec précision dans quels compartiments des entités peuvent entrer dans des modèles-jouet de virus infectant des cellules. Elles trouvent également des exclusions mutuelles (Gori et Levi 2006 ; Bodei et collab. 2015). Les analyses de dénombrement permettent aussi souvent de retrouver les invariants correspondant à la conservation du nombre de chaque sorte de protéines dans les réseaux réactionnels lorsque la composition des complexes biochimiques n’est pas représentée explicitement (Abou-Jaoudé et collab. 2015, 2016). Ces invariants sont aussi appelés invariants de places dans les réseaux de Petri.

Les modèles biologiques sont fortement concurrents et souffrent de l’explosion combinatoire dans le nombre d’entrelacements potentiels des différents événements possibles. L’interprétation abstraite a été utilisée pour oublier la séquentialité dans les traces d’exécution dans les processus de frappes (Paulevé et collab. 2011), puis plus généralement pour les réseaux asynchrones discrets booléens ou multivalués (Folschette et collab. 2013). Dans les modèles réseaux booléens ou multivalués, l’interprétation abstraite a également été utilisée pour calculer une approximation des ensembles constituant des trappes (Cook et collab. 2011 ; Klarner et collab. 2015), dans lesquels les systèmes ne peuvent plus sortir une fois entrés. Ces ensembles facilitent le calcul des trajectoires périodiques des modèles. Dans les modèles de réseaux métaboliques, l’interprétation abstraite a été utilisée pour décrire une analyse de dépendances, qui calcule l’impact potentiel de l’inhibition éventuelle d’une règle sur la concentration à l’équilibre des composants du système (John et collab. 2013 ; Allart et collab. 2019).

L’interprétation abstraite peut servir à la calibration d’un modèle (Kolcák et collab. 2019), en réalisant une partition de l’espace des paramètres en trois

régions : une première région dans laquelle le modèle satisfait une propriété temporelle donnée par l'utilisateur, une seconde qui ne la satisfait pas et une troisième pour laquelle l'analyse n'a pu conclure si la propriété était satisfaite ou non.

L'interprétation abstraite est également très utilisée pour le calcul des trajectoires des systèmes hybrides (Grosu et collab. 2011).

1.5 L'écosystème Kappa

Plusieurs outils pour analyser et manipuler des modèles Kappa sont présentés ici.

1.5.1 Analyses statiques

Un outil d'analyse statique (Boutillier et collab. 2018a), basé sur le cadre de l'interprétation abstraite, permet de calculer automatiquement certaines propriétés des modèles. Le but est d'améliorer la confiance dans les règles qui constituent le modèle. Il s'agit de retrouver des propriétés d'intérêt que le modélisateur pouvait, ou non, avoir en tête lors de la conception de son modèle ou bien de trouver des erreurs dans la modélisation.

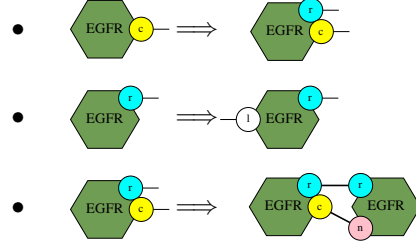
Cette analyse utilise un ensemble de motifs d'intérêt. Parmi ces motifs, l'analyse prouve que certains ne peuvent apparaître dans aucuns états potentiels du modèle. Les autres sont déclarés potentiellement accessibles : soit ils le sont effectivement, soit c'est une conséquence de la sur-approximation de l'analyse.

Les motifs d'intérêt permettent de poser les questions intéressantes sur la structure biochimique des occurrences des complexes lors de l'exécution du modèle. Actuellement, l'analyse pose trois types de questions : Existe-t-il une relation entre l'état de plusieurs sites dans les occurrences d'une protéine ? Lorsque deux occurrences de protéines sont liées entre-elles, existe-t-il une relation entre l'état de leurs sites respectifs ? Est-ce qu'une occurrence d'une protéine peut être doublement liée à une autre occurrence d'une protéine, est-ce qu'une occurrence de protéines peut être liée à des occurrences différentes d'un même type de protéines ? La première catégorie est une analyse relationnelle classique. Elle permet, par exemple, de détecter si un site ne peut être lié sans qu'un autre ne le soit ou de détecter si un site ne peut être lié sans être phosphorylé. La seconde est utile quand des sites fictifs permettent d'encoder la localisation des occurrences de protéines, il est alors possible de vérifier, chaque fois que deux occurrences de protéines sont liées, si elles se situent nécessairement dans un même compartiment. Enfin, la troisième analyse la formation de doubles liaisons entre les occurrences de protéines. Le choix exact des questions posées par l'analyseur est fixé automatiquement suite à une inspection statique des règles du modèle.

Le résultat final de l'analyse d'accessibilité est présenté à l'utilisateur sous deux formes. D'une part, les règles dont le membre gauche est en contradiction avec les motifs qui ont été prouvés inaccessibles par l'analyse sont mentionnées

à l'utilisateur. D'autre part, les propriétés intéressantes sur la structure des complexes biochimiques sont listées sous la forme de lemmes de raffinement.

Par exemple, les trois lemmes suivants :



informent l'utilisateur que (pour le premier) dans une occurrence du récepteur membranaire, le site *c* ne peut être lié sans que le site *r* ne le soit également, que (pour le second) le site *r* ne peut être lié sans que le site *l* ne le soit aussi, et que (pour le troisième) quand une occurrence du récepteur membranaire a ses sites *r* et *c* tous deux liés, ils sont nécessairement liés tous deux à une même occurrence du récepteur membranaire.

Un lemme de raffinement est ainsi présenté comme une implication entre un motif et une liste de motifs. Ici, les listes de motifs sont toutes réduites à un élément. Il faut interpréter une telle implication par le fait que toute occurrence du membre gauche de l'implication dans un état accessible peut se raffiner dans au moins un des motifs du membre droit.

Lorsque l'utilisateur obtient des propriétés auxquelles il ne s'attend pas, il doit retourner à son modèle pour comprendre l'origine du problème. Les erreurs typographiques sont assez courantes. Il arrive aussi souvent que certaines parties du modèle manquent, il faut aller les compléter ou les remplacer par des règles fictives si l'information n'est pas disponible dans la littérature. Il se peut aussi que l'état initial du modèle ait été mal choisi. Enfin, les erreurs peuvent aussi être dues à des relations causales complexes. L'analyse statique peut alors être complétée par l'analyse causale pour comprendre comment les configurations inattendues se produisent.

1.5.2 Analyses causales

La causalité est un outil très utile pour comprendre le comportement individuel des occurrences de protéines dans un modèle Kappa. Son but est d'étudier en quoi certains événements ont été nécessaires pour que d'autres événements aient pu avoir lieu.

Une trace causale est alors un ensemble d'événements dont certaines paires sont ordonnées par une relation de causalité. Celle-ci indique si l'application d'un événement a rendu possible l'application d'un autre. Un exemple de trace causale est donné en Fig. 2. Il s'agit de l'ensemble des événements pour qu'une occurrence du récepteur membranaire recrute une occurrence de la protéine *Sos* par le biais de son site *Y68*. Il faut tout d'abord activer deux occurrences du récepteur membranaire *EGFR* en les liant à des occurrences du ligand *EGF*. Les

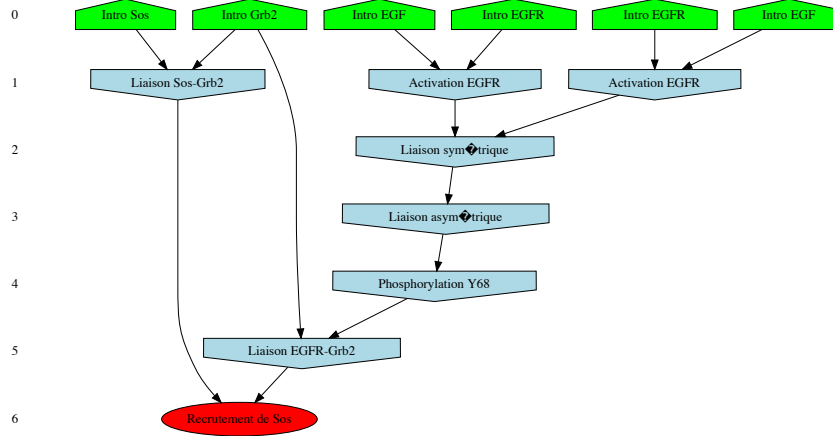


Figure 2: Une des deux traces causales pour le recrutement d’une occurrence de la protéine *Sos* par une occurrence du récepteur membranaire. Les nœuds verts représentent l’introduction des occurrences de protéines, les nœuds bleus représentent l’application des règles, le nœud rouge représente le but à observer. Les arcs décrivent les relations causales entre les événements.

deux occurrences du récepteur peuvent alors établir une liaison symétrique, puis une liaison asymétrique ce qui permet de différencier une des deux occurrences du récepteur membranaire. Le site *Y68* de cette occurrence peut alors être phosphorylé pour qu’il puisse se lier à une occurrence de la protéine de transport *Grb2*. Indépendamment, cette occurrence de la protéine de transport peut s’être liée à une occurrence de la protéine *Sos*.

Dans cette trace, tous les événements sont nécessaires, mais d’autres *scenarii* peuvent exister. Par exemple, les occurrences du récepteur membranaire peuvent recruter une occurrence de la protéine *Sos* par le biais du site *Y48*, ce qui donne lieu à une autre trace causale. Une trace causale décrit, en fait, un ensemble d’événements qui sont nécessaires dans un scénario potentiel.

Les traces causales sont obtenues en partant des résultats de la simulation, en relevant toutes les occurrences d’un événement d’intérêt. Pour chaque occurrence, une trace est extraite en collectant les événements nécessaires à cette occurrence ou récursivement à tout autre événement lui-même nécessaire (Danos et collab. 2007a). Les événements sont ensuite organisés sous la forme d’un graphe acyclique orienté grâce à la transformation de Mazurkiewicz (Mazurkiewicz 1984). Cette transformation exploite le fait que certains événements, causalement indépendants, commutent. Un moteur de recherche opérationnelle est ensuite utilisé pour retirer de cette trace causale les événements qui peuvent l’être. Une description de cette approche dans un formalisme catégorique est décrit dans cette publication (Danos et collab. 2012).

Les traces causales donnent une vision des voies de signalisation qui privilégie l’acquisition du signal. Dans un modèle, toutes les interactions sont en général réversibles, ce qui est nécessaire pour que l’occurrence d’une kinase, par exemple, puisse agir sur plusieurs occurrences de sa protéine cible à tour de rôle. Cet aspect, gestion de ressources, n’est pas du tout décrit dans les traces causales. Les traces causales ne peuvent donc pas remplacer les règles d’un modèle. Il s’agit juste d’un outil pour comprendre comment un objectif peut être atteint, mais qui ne permet pas à lui seul de définir le comportement collectif du modèle.

Les traces causales dépendent fortement de la syntaxe du langage. En effet, la syntaxe définit quelles préconditions peuvent être utilisées dans les règles, ce qui a une incidence sur le fait que deux événements puissent être vus ou non, comme indépendants causalement. Aussi, le fait que Kappa utilise de la réécriture hors contexte où seuls les sites qui ont une importance dans une interaction ont besoin d’être mentionnés, permet d’avoir plus d’événements qui commutent. Chaque trace causale peut alors résumer un plus grand nombre de traces classiques.

En Fig. 3 est considéré l’exemple d’une sorte de protéines avec deux sites de phosphorylation. Chaque site peut être phosphorylé indépendamment de l’état de l’autre site, ce qui se traduit en Kappa par les deux règles données en Fig. 3(a). Ces règles peuvent être appliquées dans n’importe quel ordre. Il y a donc une seule trace causale pour obtenir une occurrence de protéines doublement phosphorylée. Cette trace est dessinée en Fig. 3(e). Dans un réseau réactionnel, les complexes sont nommés et leur structure biochimique ne peut pas être utilisée. Il faut donc quatre réactions pour simuler ces deux règles Kappa. Or, chacune de ces réactions spécifie exactement quel réactif elle utilise, ce qui empêche les réactions de commuter. Il y a alors deux traces causales différentes selon que le site de droite ou de gauche ait été phosphorylé en premier.

Pour conclure sur la causalité, il est important de remarquer que les traces causales s’appuient sur une vision positive de la causalité. Ce n’est en général pas suffisant pour comprendre le comportement des voies de signalisation intracellulaires. En effet, il y a souvent dans ces voies des événements qui ne sont certes pas nécessaires mais qui rendent d’autres événements plus probables. C’est le cas d’une interaction qui stabiliserait une structure instable pour lui laisser le temps de réaliser une certaine interaction. D’un point de vue logique, la stabilisation de la structure n’est pas requise. Mais il est improbable que sans elle, l’autre interaction puisse avoir lieu. Ces effets cinétiques sont capturés par les notions de causalité contre-factuelles (Halpern et Pearl 2013), dont l’adaptation à Kappa (Laurent et collab. 2018) ouvre des pistes de recherches pleines de promesses.

1.5.3 Réduction de modèles

La réduction de modèles consiste à simplifier un modèle en ajustant le grain d’observation. Les réductions de modèles peuvent se formaliser comme des transformations de graphes (Gay et collab. 2014), des transformations tropicales (Radulescu et collab. 2015), des bisimulations (Camporesi et collab. 2010 ;

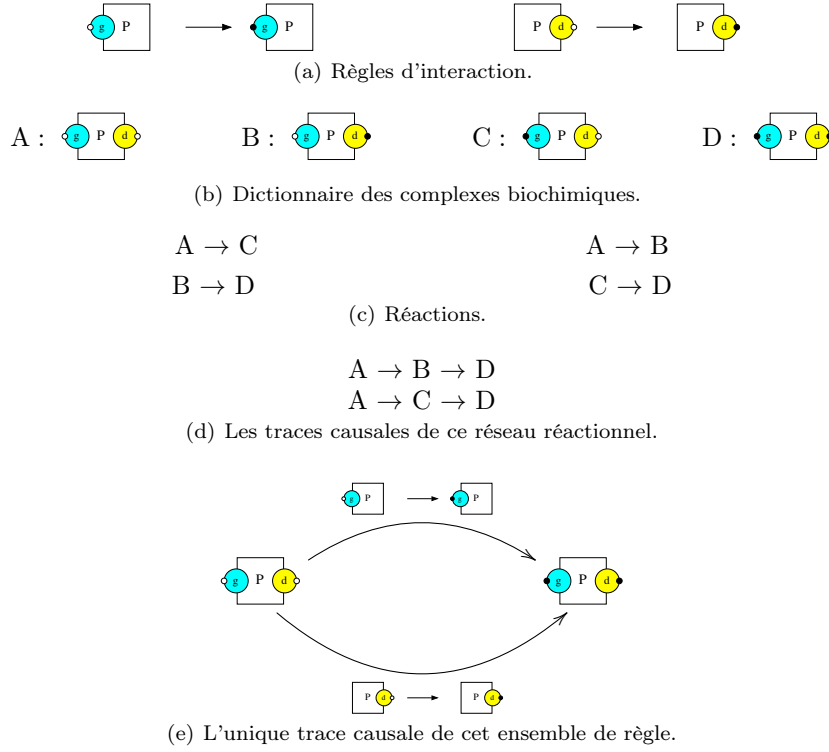


Figure 3: En 3(a), un modèle formé de deux règles d'interaction. En 3(b) l'ensemble des toutes les sortes de complexes biochimiques accessibles à partir d'une occurrence de la protéine entièrement non phosphorylée, un nom est donné à chaque sorte de complexe biochimique. En Fig. 3(c), le réseau réactionnel sous-jacent. Contrairement aux règles d'interaction, les réactions testent l'intégralité de l'état de l'occurrence de la protéine. Ainsi, les réactions qui phosphorylent les deux sites ne commutent pas. Il y a donc deux traces causales, selon que le site droit ou gauche ait été phosphorylé en premier avec les réactions (Fig. 3(d)). En Kappa, les règles de phosphorylation d'un site s'appliquent quelque soit l'état de l'autre site. Ainsi les traces causales ne distinguent pas quel site est phosphorylé en premier. Il n'y a alors qu'un seul type de trace causale (Fig. 3(e)).

Cardelli et collab. 2015), ou, tout simplement, des changements de variables (Feret et collab. 2009). Elles peuvent être classées selon la classe de propriétés qu’elles préservent.

Des outils de réduction exacte permettent de simplifier à la fois les systèmes d’équations différentielles (Feret et collab. 2009 ; Danos et collab. 2010) et les systèmes stochastiques (Feret et collab. 2013) qui sont décrit en Kappa. Ces algorithmes trouvent automatiquement des changements de variables par inspection statique des règles initiales des modèles et dérivent des modèles réduits en conséquence. La preuve de correction de ces algorithmes est faite par interprétation abstraite : le modèle réduit définit la projection exacte, par le changement de variables découvert par l’analyse, du comportement transitoire du modèle avant réduction. L’ensemble des complexes biochimiques, le changement de variables et la description extensionnelle du modèle avant réduction ne sont jamais représentés explicitement, ce qui permet à la méthode de passer à l’échelle.

Les outils de réduction de modèles pour Kappa combinent deux types d’abstraction : le premier exploite les symétries potentielles au sein des sites d’interaction des occurrences des protéines du modèle, alors que le second identifie parmi les corrélations éventuelles entre les états des sites des occurrences des protéines, celles qui n’ont aucun impact sur leur comportement collectif. Les symétries sont décrites comme des actions de groupes qui préservent l’ensemble des règles de réécriture qui constituent un modèle (Camporesi et collab. 2010 ; Feret 2015). Elles induisent une relation d’équivalence entre les complexes biochimiques qui, elle-même, définit une relation de bisimulation sur les différents états du modèle. Les états en relation seront regroupés en un seul dans le modèle réduit. Intuitivement, cette analyse détecte quels sites ont exactement les mêmes capacités d’interaction et ignore la différence entre ces sites : dans le modèle réduit, la configuration d’une occurrence d’une protéine est définie par le nombre de sites dans un certain état en faisant abstraction de quels sites précis sont dans cet état. Ceci engendre une réduction d’un facteur exponentiel : par exemple, pour un type de protéines avec n sites symétriques pouvant chacun prendre deux états différents, la réduction permet de passer de 2^n configurations potentielles à seulement $(n + 1)$.

La deuxième approche se base sur l’analyse du flot d’information entre les différents sites d’interaction des complexes biochimiques. Cela permet de comprendre quelles corrélations entre l’état des différents sites peuvent avoir une influence sur le comportement global du système et de passer les autres sous silence. Une approximation qualitative du flot d’information est calculée en répertoriant, au sein des règles de réécriture, tous les chemins entre les sites dont l’état est testé (ceux qui apparaissent dans le membre gauche d’une règle) et les sites dont l’état est modifié (ceux qui apparaissent dans le membre droit de cette règle avec un état différent de celui du membre gauche) (voir en Fig. 4(a)). Chaque motif est alors annoté en regroupant le flot d’information présent dans chacune des règles qui peut s’y appliquer. Les motifs intéressants sont ceux pour lesquels il existe un site d’interaction qui est accessible par tous les autres en suivant cette annotation. Par exemple, le complexe biochimique

dessiné en Fig. 1(a) contient les quatre motifs d'intérêt donnés en Fig. 4(b) avec leur annotation. Dans ce modèle, les motifs d'intérêt sont exactement ceux qui décrivent l'état d'un seul site $Y48$ ou $Y68$. Ainsi la corrélation entre l'état des différents sites $Y48$ et $Y68$ n'est plus représentée dans le modèle réduit. D'un point de vue combinatoire, ceci permet de passer de $m^2 \cdot n^2$ complexes biochimiques à $m+n$ motifs d'intérêt (où m et n représentent respectivement le nombre de configurations différentes pour la partie du complexe liée aux sites $Y48$ et $Y68$).

Sur un modèle plus complet (Blinov et collab. 2006 ; Schoeberl et collab. 2002 ; Brightman et Fell 2000 ; Danos et collab. 2007a), cet outil permet de passer de 10^{20} complexes biochimiques à 175,000 motifs d'intérêt, en moins de 10 minutes.

Des méthodes approchées utilisent des formes tronquées de développement formels de la sémantique stochastique (Gillespie 2009), alors que les méthodes de tropicalisation exploitent la séparation entre les échelles de temps et de concentration (Radulescu et collab. 2012). Ces méthodes ne procurent pas de bornes d'erreur explicites. Par ailleurs, elles nécessitent une description extensionnelle des réseaux réactionnels sous-jacents.

Des méthodes exactes opèrent de manière analytique pour extraire des relations d'équivalence entre les complexes biochimiques de la description explicite des réseaux réactionnels (Cardelli et collab. 2015) ou même directement sur des systèmes d'équations différentielles (Cardelli et collab. 2019). Elles permettent de calculer la meilleure bisimulation en avant, parmi celles qui sont basées sur un partitionnement des variables, et quelles variables prennent toujours la même valeur. La notion de symétries développée pour Kappa est plus restrictive car elle se concentre sur les bisimulations qui correspondent à un certain groupe de transformations. En revanche, elle permet de détecter des relations de proportionnalité entre variables. Par ailleurs, elle ne nécessite de représenter, ni les réseaux réactionnels, ni les systèmes différentiels sous-jacents, évitant ainsi un calcul dont la durée est souvent prohibitive (Camporesi et collab. 2017).

La réduction de modèles basée sur l'étude du flot d'information est à la fois une généralisation et une formalisation d'approches systématiques existantes (Borisov et collab. 2005 ; Conzelmann et collab. 2006). L'utilisation d'un langage formel et l'interprétation abstraite de sa sémantique a permis d'établir formellement la correction de ces approches.

1.6 Contenu du chapitre

Le reste du chapitre décrit le langage Kappa (Danos et Laneve 2003, 2004) sous forme graphique, ainsi que l'analyse statique qui permet de détecter quels motifs peuvent se former lors de l'exécution des modèles (Danos et collab. 2008 ; Feret et L   2018).

En particulier, la notion de graphe à sites, qui représente l'état des systèmes mod  lis  s, est introduite Sect. 2, alors que celle de r  gle de r   criture est d  crite Sect. 3. Par soucis de simplicit  , seul un fragment du langage est consid  r  . En effet, certaines constructions du langage complet font intervenir des effets

de bord (qui peuvent provoquer des transformations de l'état des occurrences de protéines, en dehors des occurrences des motifs de réécriture). S'il est possible d'adapter les différentes définitions pour traiter ces effets de bords, cela n'apporte pas grand chose conceptuellement. Par ailleurs, ce chapitre traite uniquement d'une analyse du comportement qualitatif des modèles, l'aspect quantitatif, les constantes cinétiques, ne sont pas abordées.

L'analyse statique, qui est introduite Sect. 4 permet de détecter, au sein d'un ensemble de motifs d'intérêt paramètre de l'analyse, lesquels ne peuvent jamais se former quelle que soit l'exécution du système. C'est une analyse approchée. Les motifs déclarés inaccessibles sont bien inaccessibles. Par contre, l'analyse n'apporte aucune information à propos des autres motifs. Par soucis d'efficacité, les ensembles de motifs sont organisés (Sect. 5.1) sous la forme d'une collection d'arbres de décision dans lesquels des motifs initiaux sont raffinés peu à peu en ajoutant de l'information contextuelle (Feret et Lý 2018). Cette analyse est implantée dans l'analyseur statique KaSa (Boutillier et collab. 2018a) et le choix des arbres de décisions, qui paramétrise l'analyse, est fait automatiquement par une pré-analyse. Le chapitre se conclut Sect. 6 et quelques perspectives sont données. La description du langage et de l'analyse reste volontairement assez haut niveau. Une formalisation complète et rigoureuse pour le langage complet est disponible dans les différents articles scientifiques qui sont cités dans le corps du texte.

2 Graphes à sites

La section présente décrit la notion de graphe à sites, qui permettra de représenter à la fois les différents états possibles pour les systèmes modélisés, mais aussi, les motifs qui seront utilisés dans la section 3 pour décrire, grâce à des règles de réécriture, l'évolution de l'état de ces systèmes.

2.1 Signature

En Kappa, il faut tout d'abord définir la signature des modèles. La signature d'un modèle décrit tous les ingrédients qui peuvent intervenir dans celui-ci. Elle peut être représentée graphiquement par une *carte de contacts*, comme celle dessinée en Fig. 5. Une carte de contacts comprend des nœuds pour représenter les différentes *sortes de protéines*. Ces nœuds sont nommés et adoptent des formes et des couleurs variées pour les distinguer plus facilement. Chaque sorte de protéines est associée à un ensemble de *sites d'interaction*. Ces sites sont représentés en périphérie de chaque sorte de protéines par des cercles colorés et nommés, eux-aussi. En Kappa, une sorte de protéines donnée ne peut avoir deux sites portant le même nom. Chaque site d'interaction est associé à un ensemble de pastilles colorées qui peuvent servir à représenter son *état d'activation*, comme par exemple le fait d'être – ou non – phosphorylé ou comme le fait d'être méthylé – ou non. Un état d'activation peut aussi éventuellement servir à représenter la localisation d'une occurrence d'une protéine au sein d'un en-

semble fini et fixe de compartiments cellulaires. Les sites d'interaction peuvent également porter un *état de liaison* : les sites qui portent le symbole \neg peuvent potentiellement rester libre ; la carte de contacts contient aussi des arcs non-orientés entre les sites qui peuvent potentiellement être liés deux à deux. En particulier, un site peut être lié à plusieurs sites dans la carte de contacts (il sera expliqué plus tard que de telles liaisons sont en compétition). Par ailleurs, un site peut être lié à lui-même dans une carte de contacts (il sera expliqué plus tard que ceci signifie que deux sites de deux occurrences différentes d'une même sorte de protéines peuvent être liés entre-eux).

Exemples 1 En Fig. 5 est donné un exemple de carte de contacts qui correspond aux premières interactions qui interviennent dans l'activation du facteur de croissance de l'épiderme. Cet exemple est inspiré d'un modèle BNGL disponible dans la littérature (Blinov et collab. 2004). Ce modèle a été étendu pour décrire la liaison asymétrique entre les récepteurs EGFR et traduit en Kappa. Cette carte introduit cinq sortes de protéines : des ligands EGF, des récepteurs membranaires EGFR, des protéines d'échafaudage ShC, des protéines de transport Grb2 et des protéines cibles Sos (cette dernière sera ensuite phosphorylée ce qui initiera les étapes suivantes de la cascade d'interactions). Chaque occurrence du ligand EGF a un seul site qui est nommé r ; chaque occurrence du récepteur membranaire EGFR a six sites qui sont nommés respectivement l , r , c , n , $Y48$ et $Y68$; chaque occurrence de la protéine d'échafaudage ShC dispose de deux sites qui sont nommés respectivement $Y7$ et π ; chaque occurrence de la protéine de transport Grb2 a deux sites qui sont respectivement nommés a et b ; enfin chaque occurrence de la protéine cible Sos a un seul site qui est nommé d . Seuls les sites $Y48$ et $Y68$ des occurrences de la protéine EGFR et le site $Y7$ des occurrences de la protéine ShC portent un état interne. Ces sites sont annotés par deux pastilles colorées, une blanche et une noire. La pastille blanche indique que ces sites peuvent être dans l'état non-phosphorylé, alors que la noire indique que ces sites peuvent être dans l'état phosphorylé. De plus, chaque site peut être libre (symbole \neg) ou lié. Les liaisons possibles entre sites sont entre le site r d'une occurrence de la protéine EGF et le site l d'une occurrence de la protéine EGFR ; entre les sites r de deux occurrences différentes de la protéine EGFR ; entre le site c et le n des occurrences de la protéine EGFR (il sera bientôt expliqué que la carte de contacts ne précise pas si ce doit être entre deux occurrences différentes de la protéine EGFR) ; entre le site $Y48$ d'une occurrence de la protéine EGFR et le site π d'une occurrence de la protéine ShC ; entre le site a d'une occurrence de la protéine Grb2 et le site $Y68$ d'une occurrence de la protéine EGFR ; entre le site a d'une occurrence de la protéine Grb2 et le site $Y7$ d'une occurrence de la protéine ShC (il y a donc conflit entre ces deux liaisons potentielles) ; enfin entre le site b d'une occurrence de la protéine Grb2 et le site d d'une occurrence de la protéine Sos.

2.2 Complexes biochimiques

Les modèles Kappa décrivent l'évolution d'une soupe de complexes biochimiques. Un complexe biochimique est formé de plusieurs occurrences de protéines. Chaque occurrence d'une protéine est associée à un ensemble de sites d'interaction. Chaque site peut éventuellement porter un état d'activation, mais un seul. De ce fait, si un site peut être activé de deux manières différentes, avec un état de phosphorylation et un état de méthylation par exemple, ou si un site peut être doublement activé, doublement phosphorylé par exemple, il est important de définir une pastille différente pour toutes les combinaisons potentielles d'états de ce site. Enfin, chaque site doit être soit libre, soit lié à exactement un autre site. Contrairement à la carte de contacts, un site ne peut pas être lié à lui-même dans un complexe biochimique. De plus, un site ne peut pas être lié simultanément à deux sites. Un complexe biochimique forme un graphe connexe, ce qui signifie qu'il est possible de passer de n'importe quelle occurrence de protéines à n'importe quelle autre, en suivant zéro, un ou plusieurs liens.

Exemples 2 *En Fig. 6 est donné un exemple de complexe biochimique. Ce complexe est formé de deux occurrences du ligand EGF, de deux occurrences du récepteur membranaire EGFR, de deux occurrences de la protéine d'échafaudage ShC, de deux occurrences de la protéine de transport Grb2 et d'une occurrence de la protéine Sos. Chaque occurrence du récepteur membranaire est liée au site r d'une occurrence du ligand par son site l . Les occurrences de récepteur forment un dimer grâce à une double liaison, une liaison symétrique par leurs sites r respectifs et une liaison asymétrique entre le site c de l'un et le site n de l'autre. L'occurrence du récepteur membranaire dont le site c est lié a son site Y68 phosphorylé et libre, alors que son site Y48 est phosphorylé et lié au site π d'une occurrence de la protéine d'échafaudage. Le site Y7 de cette occurrence de la protéine d'échafaudage est phosphorylé et lié au site a d'une occurrence de la protéine de transport dont le site b est lié au site d d'une occurrence de la protéine Sos. L'autre occurrence du récepteur a son site Y48 phosphorylé et lié au site π de l'autre occurrence de la protéine d'échafaudage. Le site Y7 de cette occurrence de la protéine d'échafaudage n'est ni phosphorylé, ni lié à un autre site. Enfin, le site Y68 de cette seconde occurrence du récepteur membranaire est lié au site a de l'autre occurrence de la protéine de transport. Celle-ci a son site b libre.*

La signature d'un modèle restreint l'ensemble des complexes biochimiques de ce modèle. Tous les complexes biochimiques qui sont corrects du point de vue de la syntaxe ne sont ainsi pas adéquats. Ce rôle est assuré par la carte de contacts, qui d'une part, donne la liste de tous les sites d'interaction de chaque sorte de protéines en indiquant lesquels peuvent porter un état de liaison et un état d'activation et d'autre part, résume l'ensemble des états potentiels de ces sites. Plus précisément, toute occurrence de protéines dans un complexe biochimique doit mentionner les mêmes sites que le nœud correspondant dans la carte de contacts. De plus, un site dont le site correspondant dans la carte de contacts admet au moins un état d'activation doit nécessairement avoir un

état d'activation. Il en est de même pour l'état de liaison. Ces contraintes assurent que l'état de chaque occurrence de protéines d'un complexe biochimique est entièrement défini. Trois contraintes supplémentaires assurent que l'état des sites est conforme à la carte de contacts : premièrement, un site ne peut porter un état d'activation que si le site correspondant dans la carte de contacts porte également cet état d'activation ; deuxièmement, un site ne peut être libre que si le site correspondant dans la carte de contacts peut l'être lui-aussi ; troisièmement, deux sites ne peuvent être liés que si les deux sites correspondants le sont également dans la carte de contacts. Ces trois dernières contraintes peuvent se formaliser par le fait que chaque complexe biochimique se projette sur la carte de contacts : ainsi la fonction qui associe à chaque nœud d'un complexe biochimique l'unique nœud de la même sorte dans la carte de contacts doit être un *homomorphisme*. En d'autres termes, la carte de contacts peut être vue comme un repliage de tous les complexes biochimiques du modèle et chaque nœud de la carte de contacts résume toutes les configurations possibles des protéines du type correspondant.

Exemple 1 *En Fig. 7 est représentée la projection entre le complexe biochimique dessiné dans la Fig. 6 et la carte de contacts donnée en Fig. 5. Cette projection montre que ce complexe biochimique est compatible avec cette carte de contacts.*

2.3 Motifs

L'évolution des complexes biochimiques est décrite par des règles de réécriture. Celles-ci définissent à la fois les conditions qui doivent être réalisées pour qu'une interaction donnée puisse avoir lieu et les effets potentiels de cette interaction. Avant d'expliquer ce que sont ces règles de réécriture, il est nécessaire d'expliquer la notion de motifs qui permet donc de spécifier sous quelles conditions une interaction peut avoir lieu.

Nous nous concentrons sur les motifs connexes. Des motifs plus élaborés peuvent être obtenus en juxtaposant plusieurs motifs connexes. Un motif connexe est une portion contigüe de complexe biochimique. De ce fait, il peut comporter zéro, une ou plusieurs occurrences de chaque sorte de protéines. Chaque occurrence de protéines est associée à un ensemble de sites d'interaction. Chaque site peut éventuellement porter un état d'activation. Enfin chaque site peut être libre, lié sans que le site auquel il est lié ne soit précisé ou lié exactement à un autre site (différent de lui-même donc). L'état de liaison d'un site peut également ne pas être spécifié.

Exemples 3 *En Fig. 8 est donné un exemple de motif connexe. Ce motif est formé de deux occurrences du récepteur membranaire EGFR et d'une occurrence de la protéine d'échafaudage ShC. L'occurrence de la protéine d'échafaudage mentionne uniquement son site π . Celui-ci est lié au site Y48 d'une des deux*

occurrences du récepteur membranaire. L'état d'activation de ce dernier site n'est pas précisé. Cette occurrence du récepteur membranaire mentionne également son site Y68, sans en préciser ni l'état interne ni l'état de liaison, et son site r, lui-même lié au site r de l'autre occurrence du récepteur membranaire. Cet autre occurrence mentionne également son site Y48 qui est phosphorylé mais dont l'état de liaison n'est pas spécifié et son site l qui est lié à un site qui n'est pas précisé.

Comme c'était le cas pour les complexes biochimiques, la carte de contacts contraint les motifs que l'on peut écrire dans un modèle. Ainsi, une occurrence de protéines dans un motif ne peut comporter que des sites d'interaction qui sont associés à cette sorte de protéines dans la carte de contacts. Un site ne peut porter un état d'activation que si le site correspondant dans la carte de contacts admet cet état d'activation. Un site ne peut être libre que si le site correspondant peut être libre dans la carte de contacts. Un site ne peut être lié sans préciser à quel site que si le site correspondant est lié à au moins un site dans la carte de contacts. Enfin, deux sites ne peuvent être liés ensemble que si les deux sites correspondants sont liés ensemble dans la carte de contact. En d'autres termes, comme c'était le cas pour les complexes biochimiques, il doit être possible de projeter le motif sur la carte de contacts. Cela veut dire que la fonction qui associe à chaque nœud d'un motif l'unique nœud de la même sorte dans la carte de contacts est un homomorphisme.

2.4 Plongements entre motifs

Un motif peut contenir plus ou moins d'information. En effet, il est possible d'ajouter des sites dans une occurrence de protéines qui ne mentionne pas tous ses sites. Par ailleurs, il est possible d'ajouter un état de liaison et/ou un état interne à un site qui en manque. Il est possible de préciser à quel site un site est lié quand le partenaire de celui-ci n'est pas précisé. Il est même possible de lier un site au site d'une nouvelle occurrence de protéines. Nous dirons alors que le premier motif apparaît dans le second ou encore que le second motif contient une occurrence du premier. Dans ce cas, la relation entre les occurrences de protéines du motif initial et celle du motif ainsi obtenu est formalisée par un plongement. Un plongement d'un motif vers un autre motif est une fonction qui envoie chaque occurrence de protéines du premier motif vers une occurrence de protéines du second tout en préservant la structure des graphes à sites, c'est à dire les sortes de protéines, les sites qui sont mentionnés, les états internes et les états de liaisons qui sont documentés.

Il est intéressant de remarquer que les complexes biochimiques sont des motifs connexes particuliers. Dans ces derniers, chaque occurrence de protéines décrit tous ses sites, avec un état interne et un état de liaison quand ils en ont un. Il n'est donc pas possible d'ajouter d'information dans les complexes biochimiques. Un complexe biochimique ne peut se plonger dans aucun autre motif connexe.

Exemple 2 Deux exemples de plongements sont donnés en Fig. 9. Ce sont les seuls plongements entre ce motif et ce complexe biochimique. Dans le premier (voir en Fig. 9(a)) l'unique occurrence de la protéine d'échafaudage du motif est associée à l'occurrence de la protéine d'échafaudage du complexe biochimique dont le site Y7 est libre. L'occurrence du récepteur membranaire qui est liée à l'occurrence de la protéine d'échafaudage du motif est associée à l'occurrence du récepteur membranaire qui est liée à l'occurrence de la protéine d'échafaudage dont le site Y7 est libre. Enfin, l'autre occurrence du récepteur membranaire du motif est associée à l'autre occurrence du récepteur membranaire du complexe biochimique. Il est possible de remarquer que le site l de cette dernière occurrence du récepteur est lié dans le motif, sans que le site partenaire ne soit précisé, alors qu'il est explicitement lié au site r d'une occurrence du ligand dans le complexe biochimique. Dans le second plongement (voir en Fig. 9(b)) l'occurrence de la protéine d'échafaudage du motif est associée à l'occurrence de la protéine d'échafaudage du complexe biochimique dont le site Y7 est lié. L'occurrence du récepteur membranaire qui est liée à l'occurrence de la protéine d'échafaudage du motif est associée à l'occurrence du récepteur membranaire qui est liée à l'occurrence de la protéine d'échafaudage dont le site Y7 est lié. Enfin, l'autre occurrence du récepteur membranaire du motif est associée à l'autre occurrence du récepteur membranaire du complexe biochimique.

Il est important de remarquer qu'un plongement d'un motif connexe vers un autre motif est entièrement caractérisé par l'image d'une occurrence de protéines. Pour avoir les autres associations, il suffit de suivre les liens et d'utiliser le fait qu'ils sont nécessairement préservés par le plongement. Cette propriété facilite la recherche d'occurrences de motifs dans les autres. Les graphes Kappa sont dits rigides (Danos et collab. 2010 ; Petrov et collab. 2012).

3 Réécriture de graphes à sites

Les motifs vont permettre de spécifier l'évolution potentielle de l'état des systèmes modélisés en Kappa, grâce à des règles de réécriture. C'est l'objet de cette section.

Afin de simplifier la présentation, seul un fragment du langage Kappa est présenté. En particulier, les règles de réécriture qui sont introduites dans cette section n'engendrent pas d'effets de bord. Un effet de bord est une transformation à l'extérieur du membre gauche des règles. Les effets de bords peuvent être dus à des sites libérés sans préciser à quels sites ils sont liés ou à des occurrences de protéines dégradées. Ces constructions n'ont pas été considérées afin de simplifier la présentation. Cela a permis de présenter tous les différents concepts de la syntaxe et de la sémantique de Kappa sous forme graphique.

3.1 Règles d'interaction

Les complexes biochimiques peuvent se transformer en appliquant des règles d'interaction. Une règle d'interaction est définie par une paire de motifs, qui

contiennent exactement les mêmes sortes de protéines. Le premier motif spécifie quelles conditions locales doivent être réalisées pour permettre à l'interaction de se produire. La différence entre ces deux motifs décrit quelle transformation résulte de cette interaction. Aussi le second motif d'une règle doit pouvoir être obtenu à partir du premier en changeant uniquement l'état interne et/ou de liaison de certains sites d'interaction.

Exemples 4 Des exemples de règles d'interaction sont données en Fig. 10. Celles-ci décrivent les interactions qui sont impliquées dans le recrutement des occurrences de la protéine cible par les occurrences du récepteur membranaire par leur site Y68, dans le modèle des premières étapes de l'acquisition du facteur de croissance de l'épiderme. Le recrutement par le site Y48 implique des règles d'interaction similaires, qui ne seront donc pas détaillées. La colonne de gauche décrit les interactions qui font progresser le recrutement d'une occurrence de la protéine cible. La première étape est l'activation d'une occurrence du récepteur membranaire par une occurrence du ligand (voir en Fig. 10(a)). En se liant à une occurrence du ligand, une occurrence du récepteur change de conformation et peut alors établir une liaison symétrique avec une autre occurrence du récepteur qui doit pour cela être elle-même activée (voir en Fig. 10(c)). Comme seules les occurrences du ligand peuvent se lier aux sites l des occurrences du récepteur, il n'est pas nécessaire de mentionner les occurrences du ligand dans la règle. Il suffit d'écrire que les sites l des deux occurrences du récepteur doivent être liés sans préciser à quels sites. Après cette étape, les deux occurrences du récepteur tiennent le même rôle. Pour les distinguer, une liaison asymétrique peut alors s'établir (voir en Fig. 10(e)) entre le site c d'une des deux occurrences et le site n de l'autre occurrence. Le site Y68 de l'occurrence du récepteur qui est liée par son site c peut alors se faire phosphoryler par l'autre occurrence du récepteur membranaire (voir en Fig. 10(g)). Cela change la conformation de cette occurrence du récepteur membranaire et lui permet de se lier à une occurrence de la protéine de transport (voir en Fig. 10(i)). Indépendamment, les occurrences de la protéine de transport peuvent se lier aux occurrences de la protéine cible (voir en Fig. 10(k)).

Chacune de ces interactions est réversible. Cependant les interactions inverses ne peuvent s'effectuer que sous certaines conditions. Ces interactions sont décrites dans la colonne de droite. Les liaisons symétriques entre les occurrences du récepteur membranaire capturent les occurrences du ligand qui ne peuvent alors pas se libérer (voir en Fig. 10(b)). Les liaisons asymétriques empêchent les liaisons symétriques de se briser (voir en Fig. 10(d)). Les liaisons asymétriques peuvent se briser sans condition (voir en Fig. 10(f)). La phosphorylation du site Y68 d'une occurrence du récepteur est bloquée quand ce site est lié (voir en Fig. 10(h)). Les liaisons entre les occurrences du récepteur et les occurrences de la protéine de transport d'une part, et celles entre les occurrences de la protéine de transport et celles de la protéine cible d'autre part, peuvent se défaire sans condition (voir en Figs. 10(j) et 10(l)).

Dans le langage complet, il est possible de détruire un lien entre deux oc-

currences de protéines en ne spécifiant qu'un seul des deux sites de liaisons. De plus, une règle peut également détruire des occurrences de protéines. Ces constructions peuvent induire des effets de bord, puisqu'appliquer de telles interactions est susceptible de libérer des sites qui ne sont pas décrits dans les membres gauches des règles correspondantes. Par ailleurs, le langage complet permet aussi de synthétiser de nouvelles occurrences de protéines.

3.2 Réactions induites par une règle d'interaction

Comme signalé précédemment, le membre gauche d'une règle d'interaction spécifie dans quel contexte cette interaction peut avoir lieu. Il est alors possible d'ajouter des contraintes sur les conditions d'application d'une règle en raffinant les motifs qui apparaissent dans les membres gauches et droits des règles exactement de la même manière. Une règle d'interaction qui ne peut plus être raffinée (sans ajouter de nouvelles composantes connexes) est alors appelée une règle-réaction (Harmer et collab. 2010).

Exemples 5 *En Fig. 11 est montré un exemple de deux raffinements d'une même règle d'interaction en deux règles-réactions. La règle d'interaction est celle qui permet de casser, en l'absence de lien asymétrique, le lien symétrique entre deux occurrences du récepteur membranaire (voir en Fig. 10(d)).*

1. *Dans le premier raffinement (voir en Fig. 11(a)), la règle est appliquée à un dimer dont la première occurrence du récepteur est liée par son site Y48 à une occurrence de la protéine d'échafaudage dont le site Y7 est libre et phosphorylé et par son site Y68 à une occurrence de la protéine de transport elle-même liée à une occurrence de la protéine cible. La deuxième occurrence du récepteur de ce dimer est liée par son site Y48 à une occurrence de la protéine d'échafaudage dont le site Y7 est libre et non-phosphorylé et par son site Y68 à une occurrence de la protéine de transport dont le site b est libre.*
2. *Dans le second (voir en Fig. 11(b)), les deux occurrences de la protéine d'échafaudage ont été interverties. Ainsi, la règle est appliquée à un dimer dont la première occurrence du récepteur est liée par son site Y48 à une occurrence de la protéine d'échafaudage dont le site Y7 est libre et non-phosphorylé et par son site Y68 à une occurrence de la protéine de transport elle-même liée à une occurrence de la protéine cible. La seconde occurrence du récepteur de ce dimer est liée par son site Y48 à une occurrence de la protéine d'échafaudage dont le site Y7 est libre et phosphorylé et par son site Y68 à une occurrence de la protéine de transport dont le site b est libre.*

Bien que les deux complexes biochimiques qui apparaissent dans les membres gauches de ces deux règles-réactions soient formés exactement des mêmes occurrences de protéines et dans les mêmes configurations, puisque seul l'agencement entre ces occurrences change, il apparaît que les deux règles-réactions obtenues

ne produisent pas les mêmes complexes biochimiques. Ceci justifie pleinement le choix, dans Kappa, de représenter la topologie des liens entre les occurrences de protéines. Sans celle-ci il est impossible de décrire fidèlement la séparation des occurrences de récepteurs, tout en respectant la distribution des différentes occurrences de protéines et de leurs configurations dans chacun des complexes biochimiques résultant de cette séparation. Par exemple, dans le langage BCS (Ded et collab. 2016), les complexes biochimiques sont représentés par l'ensemble des occurrences de protéines qui les constituent, ainsi que leurs configurations, mais sans préciser la topologie des liens entre ces occurrences de protéines. Aussi il est impossible de représenter fidèlement la règle de déliaison qui est dessinée en Fig. 10(d) dans ce langage.

3.3 Réseaux de réactions sous-jacents

Un ensemble de règles peut alors être traduit en un ensemble – éventuellement infini – de règles-réactions en remplaçant chaque règle d'interaction par l'ensemble des règles-réactions qui peuvent être obtenues comme raffinement de ces règles.

Ensuite, quitte à nommer les différents complexes biochimiques qui peuvent intervenir dans les règles-réactions ainsi obtenues, nous pouvons assimiler ces règles-réactions à un réseau de réactions (éventuellement infini), dans lequel chaque réaction est spécifiée par une liste de réactifs et une liste de produits parmi un ensemble d'espèces biochimiques représentées uniquement par des noms (en passant sous silence leurs structures biochimiques). Ce réseau de réactions est défini de manière unique modulo le choix des noms associés aux espèces biochimiques.

Exemples 6 Pour conclure cette section, nous détaillons la génération d'un réseau de réactions à partir d'un ensemble jouet de règles Kappa. Nous considérons un modèle avec une seule sorte de protéines qui admet deux sites, g et d , chacun pouvant être phosphorylé ou non. La signature du modèle est donnée par la carte de contacts qui est dessinée en Fig. 12(a). Le phosphorylation et la déphosphorylation de chaque site dans une occurrence de protéines peut se faire indépendamment de l'état de l'autre site, ce qui est formalisé dans les quatre règles données en Fig. 12(b). Ainsi ni les règles de phosphorylation, ni celles de déphosphorylation d'un site, ne mentionnent l'état de phosphorylation de l'autre site.

Pour obtenir les règles-réactions associées à ce modèle jouet, il suffit d'explicitier dans quel contexte local les interactions peuvent se produire. Ainsi chaque règle Kappa donne ici lieu à deux règles-réactions selon que le site qui n'est pas mentionné dans la règle initiale est phosphorylé ou non. Ces règles-réactions sont données en Fig. 12(c).

La prochaine étape est de nommer les différents complexes biochimiques qui interviennent dans les règles-réactions ainsi obtenues. Une occurrence de protéines dont aucun site n'est phosphorylé, est appelée A , une occurrence de protéines dont seul le site d est phosphorylé, est appelée B , une occurrence de

protéines dont seul le site g est phosphorylé, est appelée C et une occurrence de protéines dont les deux sites sont phosphorylés, est appelée D . Les réactions données en Fig. 12(e) sont obtenues en remplaçant chaque occurrence de complexe biochimique par son nom dans les règles-réactions.

Le choix d’une sémantique en terme de réseaux de réactions a été fait pour simplifier la présentation. C’était ainsi que le langage BNGL avait été implanté initialement (Blinov et collab. 2004). Une telle sémantique est toutefois assez peu utile en pratique, car un modèle Kappa engendre en général un trop grand nombre de réactions. Par contre, la sémantique de Kappa peut être formalisée directement, soit sous forme d’une algèbre de processus (Danos et collab. 2008 ; Feret et collab. 2013), soit dans un cadre catégorique (Danos et collab. 2012 ; Feret 2015). La première méthode est plus opérationnelle alors que la seconde abstrait au contraire beaucoup de détails. Il faut cependant noter que les cadres catégoriques usuels de la réécriture de graphes, que ce soit par push-out simples (Löwe 1993), push-out doubles (Corradini et collab. 1997) ou sesqui-pushout (Corradini et collab. 2006)) ne représentent pas fidèlement les effets de bord avec la définition usuelle des plongements entre graphes à sites. Deux approches connues permettent d’y remédier. Il est possible soit de changer la définition des plongements (Danos et collab. 2012 ; Feret 2015), soit d’enrichir les objets de la catégorie par des contraintes (Behr et Krivine 2019).

La simulation d’un modèle Kappa opère directement par réécriture du graphe qui représente l’état du système, sans avoir à considérer le réseau de réactions sous-jacent (Danos et collab. 2007b ; Boutillier et collab. 2017).

4 Analyse des motifs accessibles

Si la carte de contacts (e.g. voir en Fig. 5 à la page 52) donne un aperçu rapide de toutes les interactions potentielles entre les différents sites des occurrences des protéines dans un modèle, elle n’est en générale pas suffisante pour décrire précisément la structure de ses complexes biochimiques. En effet, l’état des différents sites d’interaction d’un complexe biochimique est souvent contraint par des invariants structurels. Par exemple, dans le modèle des premières étapes de l’acquisition du facteur de croissance de l’épiderme, les sites $Y48$ et $Y68$ des occurrences du récepteur membranaire, ainsi que le site $Y7$ des occurrences de la protéine d’échafaudage, ne peuvent être liés à un autre site sans être phosphorylés (à moins que ce soit le cas dans l’état initial). Par ailleurs, lorsque les deux sites r et c d’une occurrence du récepteur sont liés simultanément, ils sont nécessairement liés respectivement au site r et au site n d’une même occurrence du récepteur (ce qui forme une double liaison). Un autre exemple concerne les modèles avec des compartiments, comme, par exemple, une cellule dont on distingue le noyau du cytoplasme. La localisation de chaque occurrence de protéines peut alors être spécifiée comme l’état d’activation d’un site fictif. Dans de tels modèles, toutes les occurrences de protéines d’un même complexe biochimique sont en général localisées dans un même compartiment, ce qui se

traduit par la contrainte que le site fictif de deux occurrences de protéines liées entre elles doit toujours être dans le même état. Dans certains cas, il est toutefois possible d’avoir des complexes transmembranaires avec des portions localisées dans des compartiments voisins, c’est à dire de part et d’autre d’une membrane.

Dans cette section est décrite une analyse statique qui permet de détecter automatiquement ces contraintes, afin de vérifier que les propriétés auxquelles peut s’attendre le modélisateur sont bien vérifiées ou bien de détecter certaines erreurs de modélisation. En particulier, cette analyse permet de trouver des *règles mortes*. Ce sont des règles qui ne peuvent jamais s’appliquer dans un modèle, car les contraintes qui sont exprimées dans leurs membres gauches ne sont pas réalisables. C’est souvent la conséquence d’erreurs typographiques (par exemple, quand une même sorte de protéines est désignée par deux noms différents dans l’encodage d’un modèle), d’un état initial incomplet, d’interactions manquantes dans le modèle (par exemple, quand l’activation d’un site n’est pas décrite, alors qu’elle est nécessaire pour la suite de la cascade d’interactions) ou de conditions causales plus complexes qu’il faut alors élucider.

Cette analyse est implantée dans l’analyseur statique KaSa (Boutillier et collab. 2018a) et intégrée dans la plate-forme de modélisation en ligne dédiée au langage Kappa (Boutillier et collab. 2018b). Ceci permet d’assister le modélisateur pendant l’écriture du modèle en lui fournissant les contraintes structurelles qui sont vérifiées par les complexes biochimiques et en l’avertissant de la présence de règles mortes, après chaque ajout ou modification d’une règle d’interaction.

4.1 Accessibilité dans un réseau réactionnel

La première étape consiste à définir l’ensemble des états accessibles dans un modèle Kappa. Comme nous l’avons vu dans la section 3.3 page 24, un modèle Kappa induit un réseau réactionnel, ce qui permet de définir directement l’ensemble des états accessibles d’un modèle Kappa sans recourir à des constructions compliquées.

Soit un réseau réactionnel, c’est à dire un ensemble d’espèces biochimiques \mathcal{S} et un ensemble de réactions \mathcal{R} . Chaque réaction est donnée par deux listes d’espèces biochimiques : ses réactifs et ses produits. Ce réseau induit un système de transitions dans lequel l’état du réseau est défini comme un certain nombre (éventuellement nul) d’occurrences de chacune des espèces biochimiques – c’est à dire une fonction de l’ensemble \mathcal{S} vers l’ensemble \mathbb{N} des entiers naturels — et les *transitions* permettent de sauter d’un état à un autre en consommant les réactifs d’une réaction et en ajoutant les produits de cette même réaction (en tenant compte de leur multiplicité). Une transition n’est possible que si l’état courant du système contient tous les réactifs qui sont nécessaires à la réaction (en tenant compte, une nouvelle fois, de leur multiplicité). Une transition d’un état q vers un autre état q' est alors notée $q \rightarrow q'$.

Exemple 3 *Un système de transitions est donné comme exemple en Fig. 13(b). Il correspond à la restriction du système de transitions associé aux réactions qui sont données en Fig. 13(a) aux états accessibles à partir d’un état initial formé*

de six occurrences de l'espèce biochimique A . Dans ce réseau, la somme entre le nombre d'occurrences de A , de B et deux fois celui de C est toujours égal à la quantité initiale de A . En effet, cette quantité n'est modifiée par l'application d'aucune des réactions du réseau.

Étant donné un ensemble d'états initiaux potentiels, $\mathcal{I} \subseteq \mathcal{S}^{\mathbb{N}}$, nous définissons l'ensemble des états accessibles comme étant ceux susceptibles d'être atteints à partir d'un état initial (de l'ensemble \mathcal{I}) en appliquant un nombre arbitraire (éventuellement nul) de transitions. Cet ensemble peut se définir comme le plus petit point-fixe de la fonction suivante :

$$\mathbb{F} : \begin{cases} \wp(\mathcal{S}^{\mathbb{N}}) & \rightarrow \wp(\mathcal{S}^{\mathbb{N}}) \\ X & \rightarrow \mathcal{I} \cup \{q' \mid \exists q \in X, q \rightarrow q'\} . \end{cases}$$

Il faut noter que la fonction \mathbb{F} est croissante, ce qui signifie que si X_1 et X_2 sont deux ensembles d'états tels que l'ensemble X_1 soit un sous-ensemble de l'ensemble X_2 , alors l'ensemble $\mathbb{F}(X_1)$ est nécessairement un sous-ensemble de l'ensemble $\mathbb{F}(X_2)$ lui-aussi. Comme, de plus, cette fonction est définie sur l'ensemble des parties d'un ensemble, le *théorème de Tarski* (Tarski 1955) assure que la fonction \mathbb{F} admet un point fixe, plus petit que tout autre point fixe de \mathbb{F} . Ce plus-petit point fixe, que l'on note *lfp* \mathbb{F} , est en fait l'ensemble des états accessibles.

Malheureusement, le calcul de ce plus petit point fixe peut être coûteux, voire ne pas terminer. Ceci motive la construction d'abstractions pour calculer un sur-ensemble des états accessibles en un temps de calcul acceptable.

4.2 Abstraction d'un ensemble d'états

Lorsqu'un réseau est induit par un modèle Kappa, la structure biochimique associée aux espèces de ce réseau peut être utilisée pour construire une abstraction. Une possibilité consiste à choisir un ensemble de motifs connexes afin d'abstraire les ensembles d'états par le sous-ensemble parmi ces motifs de ceux qui apparaissent au moins une fois dans au moins un état de cet ensemble. Le choix des motifs connexes considérés est important : il définit le compromis entre l'expressivité de l'abstraction, c'est à dire son niveau d'approximation, et sa complexité, c'est à dire le coût pour effectuer des calculs à ce niveau d'abstraction.

Exemple 4 *Un exemple de motifs d'intérêt pour le modèle des premières interactions de l'acquisition du facteur de croissance de l'épiderme est donné en Fig. 14. Les huit premiers motifs se concentrent sur l'analyse des relations potentielles entre l'état des sites l , r et c dans les occurrences du récepteur membranaire. Ils correspondent à toutes les combinaisons syntaxiquement possibles pour l'état de liaison de ces 3 sites. Ce sont des vues locales (ou plus précisément des sous-vues locales) (Danos et collab. 2008). Elles permettent d'abstraire un ensemble de complexes biochimiques par l'ensemble de toutes les configurations potentielles de toutes ses occurrences de protéines, vues indépendamment*

les unes des autres. Ceci revient à garder uniquement l'information à propos de l'état de liaison et l'état d'activation de chaque site dans chaque occurrence de protéines tout en passant sous silence à quel site chaque site lié l'est.

La formation de dimers dans ce modèle fait intervenir des doubles liaisons. Il est légitime de se demander s'il est possible de former des chaînes comportant successivement au moins trois occurrences du récepteur membranaire. C'est le but des deux derniers motifs de l'ensemble. Ils permettent de distinguer le cas d'une double liaison entre deux occurrences du récepteur de celui de trois occurrences du récepteur liées consécutivement, en s'interrogeant pour chaque occurrence du récepteur membranaire dont les sites r et c sont liés, si elle peut être liée à une même occurrence du récepteur ou si elle peut être liée à deux occurrences différentes. En toute rigueur, pour s'assurer qu'une chaîne d'au moins trois occurrences du récepteur ne peut pas se former, il faut également considérer des motifs d'intérêt similaires pour la paire de sites r et n et la paire de sites c et n .

Plus précisément, l'abstraction est paramétrée par le choix d'un ensemble \mathcal{P} de motifs connexes. L'ensemble \mathcal{P} regroupe des motifs d'intérêt, ainsi que des motifs qui seront utilisés de manière intermédiaire dans la preuve que certains de ces motifs d'intérêt sont inaccessibles. Une sous-partie de l'ensemble \mathcal{P} est appelée une propriété abstraite. Chaque propriété abstraite représente un ensemble d'états concrets : un état concret q sera dit compatible avec une propriété abstraite X^\sharp si et seulement si aucun motif qui est dans l'ensemble \mathcal{P} sans être dans l'ensemble X^\sharp n'apparaît dans un complexe biochimique présent dans l'état q . L'ensemble de tous les états concrets compatibles avec la propriété abstraite X^\sharp est alors noté $\gamma_{\mathcal{P}}(X^\sharp)$. Qui peut le plus, peut le moins : plus nombreux sont les motifs autorisés, plus nombreux sont les complexes biochimiques compatibles. La fonction $\gamma_{\mathcal{P}}$ est donc croissante. Elle permet de définir formellement la notion d'abstraction d'un ensemble d'état : une propriété abstraite X^\sharp sera dite être une abstraction d'un ensemble d'état X si et seulement si l'ensemble X est inclus dans l'ensemble $\gamma_{\mathcal{P}}(X^\sharp)$. La fonction $\gamma_{\mathcal{P}}$ est couramment appelée la *fonction de concrétisation*. De plus l'image d'une propriété abstraite par cette fonction, est appelée sa *concrétisation*.

Réciproquement, étant donné un ensemble d'états X , l'ensemble des éléments de l'ensemble \mathcal{P} qui apparaissent dans au moins un complexe biochimique d'un état élément de l'ensemble X sera noté $\alpha_{\mathcal{P}}(X)$. La fonction $\alpha_{\mathcal{P}}(X)$ est croissante également. La propriété abstraite $\alpha_{\mathcal{P}}(X)$ est en fait la *meilleure approximation* de l'ensemble d'états X , ce qui signifie que d'une part c'est une abstraction de l'ensemble X (i.e. $X \subseteq \gamma_{\mathcal{P}}(\alpha_{\mathcal{P}}(X))$) et que d'autre part c'est un sous-ensemble de toute autre abstraction de X (i.e. pour tout sous-ensemble Y de l'ensemble \mathcal{P} tel que $X \subseteq \gamma_{\mathcal{P}}(Y)$, l'inclusion $\alpha_{\mathcal{P}}(X) \subseteq Y$ est vérifiée). La paire de fonctions $(\alpha_{\mathcal{P}}, \gamma_{\mathcal{P}})$ est alors appelée une *correspondance de Galois* (Cousot et Cousot 1977 ; Cousot 1999).

Exemple 5 En Fig. 15 est introduit un exemple jouet pour mieux comprendre le comportement des fonctions d'abstraction et de concrétisation. La signature

de ce modèle peut être consultée en Fig. 15(a). Il existe une seule sorte de protéines, qui est appelée *A*. Cette protéine est munie de deux sites *g* et *d* (pour gauche et droite). La carte de contacts spécifie que chaque site peut être libre et qu'un site *g* peut être lié à un site *d* d'une même ou d'une autre occurrence de la protéine *A*. L'abstraction induite par l'ensemble des motifs d'intérêt donné en Fig. 15(b) repose sur les vues locales des occurrences de cette protéine. Elle permet de se poser la question de l'existence ou non, d'une relation entre l'état de liaison des sites *g* et *d* dans chaque occurrence de la protéine *A*.

Exemple 6 Des exemples de meilleures approximations sont donnés en Fig. 15(c). Par abus de langage, nous appelons la meilleure approximation d'un complexe biochimique, la meilleure approximation de l'ensemble formé d'un seul état lui-même formé de ce seul complexe. Le modèle admet deux types de complexes biochimiques. Les occurrences de la protéine *A* peuvent former des chaînes d'occurrences de protéines liées successivement par leur site *d* et *g*, laissant le site *g* de la première occurrence de la protéine *A* et le site *d* de la dernière occurrence de la protéine *A* libres. Les occurrences de la protéine *A* peuvent aussi former des anneaux en reliant le premier et le dernier sites d'une chaîne d'occurrences de protéines. La meilleure approximation d'une chaîne d'occurrences de la protéine *A* dépend de la taille de cette chaîne. Ainsi, la meilleure approximation d'une chaîne réduite à une occurrence de la protéine *A* est l'ensemble qui contient uniquement la vue locale dont les deux sites sont libres ; la meilleure approximation d'une chaîne formée d'exactly deux occurrences de la protéine *A* est l'ensemble qui contient deux vues locales : l'une avec le site *g* libre et le site *d* lié, l'autre avec le site *g* lié et le site *d* libre ; enfin la meilleure approximation des chaînes d'occurrences de la protéine *A* de longueur au moins égale à 3 contient également la vue locale dont les deux sites *g* et *d* sont liés. Par contre, la meilleure approximation d'un anneau d'occurrences de la protéine *A* est toujours l'ensemble formé uniquement de la vue locale dont les deux sites *g* et *d* sont liés, et ce, quelle que soit la longueur de cette anneau. La fonction qui associe à chaque ensemble de complexe biochimique sa meilleure approximation est distributive. Cela signifie que la meilleure approximation d'un ensemble de complexes biochimiques est l'union de la meilleure approximation des singletons correspondants.

Exemple 7 Des exemples de concrétisations sont donnés en Fig. 15(d). Par définition, la concrétisation de l'ensemble formé uniquement de la vue locale dans laquelle les deux sites sont libres, est l'ensemble de tous les états qui ne contiennent pas d'autres vues locales. Il s'agit donc des états qui ne contiennent que le complexe biochimique composé d'exactly une occurrence de la protéine *A*. Par la même démarche, la concrétisation de l'ensemble formé uniquement de la vue locale dont les deux sites sont liés est l'ensemble de tous les états qui ne contiennent que des anneaux d'occurrences de la protéine *A* (quitte à utiliser cette vue locale plusieurs fois). Par contre, tout complexe contenant une vue locale avec exactly un site libre, doit contenir également une vue locale avec l'autre site, libre. De ce fait, la concrétisation d'un ensemble composé d'une

seule vue locale avec un site libre et l'autre lié, est l'ensemble ne contenant que l'état vide (qui est noté \emptyset). Si seules les deux vues locales où un site est lié et l'autre libre sont autorisées, seules des chaines de deux occurrences de la protéine A peuvent être construites. Enfin, si seule la vue locale avec les deux sites libres est interdite, il est possible de former n'importe quelle chaîne d'occurrences de la protéine A de taille au moins égale à 2 et n'importe quel anneau d'occurrences de protéines (sans restriction de taille). La fonction de concrétisation n'est pas distributive (l'image de l'union de deux ensembles de vues locales peut être un sur-ensemble strict de l'union de leurs images).

Les fonctions $\alpha_{\mathcal{P}}$ et $\gamma_{\mathcal{P}}$ se composent dans les deux sens. Ces compositions sont révélatrices des traits principaux du choix de l'abstraction. La composée $\gamma_{\mathcal{P}} \circ \alpha_{\mathcal{P}}$ caractérise le niveau d'approximation. En effet, pour tout ensemble d'états X , $\gamma_{\mathcal{P}}(\alpha_{\mathcal{P}}(X))$ est le plus grand ensemble d'état qui a la même meilleure approximation que X . Il est impossible ainsi de distinguer ces deux ensembles en terme de propriétés abstraites. En revanche, la composée $\alpha_{\mathcal{P}} \circ \gamma_{\mathcal{P}}$ témoigne d'une certaine combinatoire dans le domaine abstrait. Elle associe à chaque propriété abstraite, la plus petite propriété abstraite qui est satisfaite par le même ensemble d'états concrets. Appliquer cette composée permet donc de raffiner une propriété abstraite, par déduction, et ce sans perdre le moindre état concret.

Exemple 8 Appliquée à l'ensemble formé d'un seul état composé uniquement d'un anneau de taille 1, la composée de fonctions $\gamma_{\mathcal{P}} \circ \alpha_{\mathcal{P}}$ donne l'ensemble des états formés uniquement d'anneaux d'occurrences de la protéine A . En effet, la meilleure approximation d'un anneau de taille 1, est l'ensemble de vues locales composé uniquement de la vue dont les deux sites sont liés. Or, voir également en Fig. 15(d), la concrétisation de cet ensemble de vues locales est l'ensemble de tous les états formés uniquement d'anneaux. Ainsi le niveau d'abstraction ne permet de distinguer, ni le nombre d'occurrences, ni la taille des anneaux d'occurrences de la protéine A .

Exemple 9 Appliquée à l'ensemble formé exactement des deux vues locales, la première avec le site g libre et le site d lié, la seconde avec les deux sites liés, la composée de fonctions $\alpha_{\mathcal{P}} \circ \gamma_{\mathcal{P}}$ retourne l'ensemble formé d'une seule vue locale, celle avec les deux sites liés. En effet, la première vue ne peut apparaître dans un état sans que celui-ci ne contienne une occurrence de la vue locale avec le site d libre et le site g lié. De ce fait, elle ne peut apparaître dans aucun état de la concrétisation de l'ensemble formé par ces deux vues locales et n'est donc pas un élément de la meilleure approximation de l'ensemble de ces états. Ainsi, un état abstrait peut contenir des motifs d'intérêt, qui ne peuvent apparaître dans des complexes biochimiques sans contenir des occurrences de motifs d'intérêt interdits. Retirer ces motifs ne change pas l'ensemble des états concrets qui satisfont la propriété abstraite, mais cette étape peut requérir un temps de calcul substantiel.

4.3 Transferts de point-fixes

Le plus petit point fixe qui définit l'ensemble des espèces biochimiques accessibles dans un réseau réactionnel, pour un état initial donné, peut se calculer au niveau des propriétés abstraites grâce à la correspondance de Galois $(\alpha_{\mathcal{P}}, \gamma_{\mathcal{P}})$.

Pour cela, il faut tout d'abord construire la *contre-partie abstraite* de la fonction \mathbb{F} , qui agira, non pas sur des ensembles d'états concrets, mais directement sur les propriétés abstraites. Cette contre-partie abstraite se définit de manière systématique : il suffit, pour chaque propriété abstraite X^\sharp , de considérer l'ensemble des états concrets $\gamma_{\mathcal{P}}(X^\sharp)$ qui vérifient la propriété X^\sharp , puis d'appliquer la fonction \mathbb{F} à cet ensemble et enfin d'appliquer à ce résultat la fonction $\alpha_{\mathcal{P}}$ pour en calculer la meilleure approximation. C'est même la manière correcte la plus précise de procéder : la fonction $\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}$ est, en effet, la meilleure contre-partie abstraite de la fonction \mathbb{F} (Cousot et Cousot 1979). Elle permet de déléguer le calcul des états accessibles au domaine abstrait en contre-partie d'une perte éventuelle de précision. Pour ce faire, il suffit de remarquer que la fonction $\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}$ est croissante (comme composée de fonctions croissantes) et définie sur l'ensemble des parties d'un ensemble. Elle admet donc un plus petit point fixe qui sera noté $lfp(\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}})$. L'inclusion suivante : $lfp \mathbb{F} \subseteq \gamma_{\mathcal{P}}(lfp(\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}))$ se prouve alors par induction (Cousot et Cousot 1979). Autrement dit le plus petit point fixe de la fonction $\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}$ est une abstraction de l'ensemble des états accessibles du modèle considéré. C'est à dire que la propriété abstraite $lfp(\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}})$ est satisfaite par chaque état accessible du modèle.

Le calcul des itérations de la fonction $[\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}]$ peut prendre beaucoup de temps. Il est possible d'ajuster le compromis entre précision et temps de calcul en remplaçant celle-ci par une fonction moins précise. En effet, pour toute fonction \mathbb{F}^\sharp telle que $[\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}](Y) \subseteq \mathbb{F}^\sharp(Y)$ pour tout ensemble de motifs $Y \subseteq \mathcal{P}$, l'inclusion $lfp \mathbb{F} \subseteq \gamma_{\mathcal{P}}(lfp(\mathbb{F}^\sharp))$ est également satisfaite (Cousot et Cousot 1979).

Une telle fonction \mathbb{F}^\sharp peut être dérivée à la main. Pour cela, il faut d'abord donner une définition plus explicite de la fonction $[\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}]$. Appliquée à un sous-ensemble $Y \subseteq \mathcal{P}$ de motifs d'intérêt, cette fonction ajoute l'ensemble des nouveaux motifs d'intérêt qui peuvent apparaître dans un état accessible en une étape de réécriture à partir d'un état qui ne contient aucun motif de l'ensemble \mathcal{P} qui ne serait pas dans l'ensemble de motifs Y . Or, une telle étape de réécriture est nécessairement induite par une règle-réaction, elle-même induite par une règle du modèle. Chaque nouveau motif P doit donc apparaître dans le membre droit d'une règle-réaction et l'ensemble d'états singleton formé du membre gauche de cette règle-réaction ne doit contenir aucune occurrence de motifs de l'ensemble $\mathcal{P} \setminus Y$. Dans cette règle-réaction, l'occurrence du motif P dont il est question et l'image du membre droit de la règle sous-jacente ont nécessairement au moins une occurrence de protéines en commun (sinon le motif P apparaîtrait également dans le membre gauche de la règle-réaction et ne serait donc pas un nouveau motif). Il est alors possible de fixer le motif P au membre droit de cette règle, en unifiant les occurrences de protéines du motif P et de

la règle du modèle qui sont communes dans la règle-réaction. Ceci forme alors un raffinement du membre droit de la règle. Un raffinement de la règle peut alors être construit en ajoutant toute information présente dans le motif P qui n'est pas déjà présente dans le membre droit initial de la règle, dans le membre gauche de la règle. Le résultat est une spécialisation de la règle à la production du motif P à cette position particulière. Par construction, le membre gauche de la règle raffinée apparaît dans un état dans la concrétisation de Y .

Ainsi, pour calculer les nouveaux motifs d'intérêt de l'ensemble $[\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}](Y)$, il suffit de calculer tous les *chevauchements* possibles entre un nouveau motif d'intérêt potentiel (dans $\mathcal{P} \setminus Y$) et un membre droit d'une règle du modèle. Chaque chevauchement induit un raffinement de la règle correspondante. Si le membre gauche de la règle raffinée apparaît dans un état de l'ensemble $\gamma_{\mathcal{P}}(Y)$, alors ce motif appartient bien à l'ensemble $[\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}]Y$.

Exemples 7 *Un exemple de cette construction est dessiné en Fig. 17. L'état abstrait actuel est donné en Fig. 17(a). Seules trois vues locales sont pour l'instant autorisées, celle avec aucun des sites l , r , c lié, celle avec seul le site l lié et celle avec seuls les sites l et r liés. Par ailleurs, ni les doubles liaisons, ni les chaînes d'au moins trois récepteurs ne sont autorisées à cet instant de l'analyse. La preuve que l'on peut construire une occurrence de protéines avec les trois sites l , r et c liés, est donnée en Fig. 17(b). Il suffit d'identifier cette vue locale à l'occurrence gauche du récepteur dans le membre droit de la règle (les plongements en pointillés bleus et marrons envoient ces deux occurrences de protéines sur une même occurrence de protéines) qui permet d'établir une liaison asymétrique entre deux récepteurs membranaires (voir en Fig. 10(e)). Ceci est possible car les sites en commun dans ces deux occurrences de protéines sont dans un état compatible : en effet, la vue locale demande que ces sites soient liés, alors que le membre droit de la règle précise à quels sites ils le sont. La règle est alors spécialisée à la production de la nouvelle vue locale pour ce chevauchement particulier entre la vue locale et le membre droit de la règle. Cela consiste à ajouter dans les membres gauche et droit de la règle l'information que le site l est lié. Pour conclure, il suffit alors d'exhiber un plongement (ici dessiné en pointillés rouges) en le membre gauche de la règle ainsi raffinée et une espèce biochimique, en vérifiant que cette espèce biochimique ne contient aucune occurrence des motifs d'intérêt non encore autorisés. Le dimer avec uniquement une liaison symétrique et dans lequel les deux occurrences du récepteur membranaire sont toutes deux liées à des occurrences du ligand et les sites $Y48$ et $Y68$ non phosphorylés et libres rempli parfaitement ces conditions. Ainsi la vue locale dans laquelle les trois sites l , r et c sont liés simultanément sera utilisable dès la prochaine itération de l'analyse.*

Lors du calcul de l'ensemble $[\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}](Y)$, l'étape la plus coûteuse en temps de calcul est de vérifier que les membres gauches des règles raffinées peuvent apparaître dans un état de l'ensemble $\gamma_{\mathcal{P}}(Y)$. La section suivante a pour but de réduire ce coût moyennant une approximation supplémentaire.

5 Analyse par ensembles de motifs orthogonaux

Ajouter des hypothèses sur l'ensemble des motifs d'intérêt et simplifier le test de réalisabilité du membre gauche des raffinements de règles en le remplaçant par une condition nécessaire, mais pas toujours suffisante, permet de rendre ce calcul plus efficace au prix d'une perte de précision de l'analyse. Ceci permet de définir une approximation correcte de la fonction $\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}$.

5.1 Ensembles de motifs orthogonaux

Pour ce faire, l'ensemble des motifs d'intérêt peut être organisé sous la forme d'un ensemble fini d'ensembles finis de motifs orthogonaux (Feret et Lý 2018). Chaque *ensemble de motifs orthogonaux* est un arbre de décision raffinant progressivement un motif initial, dans le but de répondre à une question spécifique. Un ensemble de motifs orthogonaux est construit de manière à ce que toute occurrence du motif initial dans une espèce biochimique, puisse être complétée en exactement une occurrence d'un de ces raffinements. En conséquence, les raffinements du motif initial sont deux à deux incompatibles et ils recouvrent, en quelque sorte, tous les cas possibles pour le motif initial.

Le choix exact des ensembles de motifs orthogonaux repose sur une analyse préliminaire qui calcule, par inspection des règles du modèle, quelles questions intéressantes se posent. Trois catégories de questions sont considérées par défaut dans l'analyseur KaSa (mais il est possible de paramétrer l'analyse pour en désactiver une ou deux). La première infère des relations entre les états des différents sites de chaque type de protéines, cela correspond à l'analyse des vues locales (Danos et collab. 2008). La seconde permet de détecter des relations entre l'état des sites dans des occurrences de protéines qui partagent un lien (Feret et Lý 2018) dans le but d'analyser les déplacements de complexes biochimiques lorsque ceux-ci sont codés par des transformations de l'état d'activation de sites fictifs. L'analyse permet alors de vérifier si oui ou non deux occurrences de protéines sont toujours localisées dans le même compartiment quand elles sont liées entre elles. La troisième permet de détecter si une même occurrence de protéines peut être liée simultanément à deux occurrences différentes de protéines ou si une même occurrence de protéines peut être liée au moins doublement à une autre occurrence de protéines (Feret et Lý 2018). Une quatrième sorte d'ensembles de motifs orthogonaux est en cours d'implantation. Elle se concentre sur la formation de complexes biochimiques cycliques : son but est de prouver l'absence de complexes biochimiques de taille non bornée (Boutillier et collab.).

Les ensembles finis de motifs orthogonaux peuvent être construits récursivement, en remplaçant un des motifs par plusieurs motifs le raffinant. Il suffit de choisir une information non spécifiée dans ce motif et de considérer tous les cas possibles pour cette information, d'où la représentation sous forme d'arbre de décision. L'ensemble de motifs orthogonaux est alors formé par les feuilles de cet arbre, alors que les nœuds de cet arbre représentent les motifs intermédiaires qui ont été remplacés par des motifs plus précis.

Exemple 10 *L'ensemble des motifs d'intérêt introduit en Fig. 14 est inclus dans la réunion de deux ensembles de motifs orthogonaux. En effet, l'ensemble des vues locales peut être obtenu, en partant d'une occurrence de la protéine EGFR sans aucun site, en se demandant successivement si le site l est libre ou non, si le site r est libre ou non et si le site c est libre ou non. L'arbre de décision correspondant se trouve en Fig. 18(a). Les deux derniers motifs d'intérêt sont obtenus en se demandant si un récepteur peut établir des liaisons doubles. Partant d'une occurrence de la protéine EGFR sans aucun site, il faut se demander si le site r est libre ou non, puis dans le cas où le site r est lié, si le site c est lié ou non, et enfin, dans le cas où le site c est également lié, si ces deux sites sont liés à une même occurrence de récepteur membranaire ou à deux occurrences différentes. L'arbre de décision ainsi obtenu est donné en Fig. 18(b).*

Les différents ensembles de motifs orthogonaux collaborent au sein de l'analyse, qui effectue ainsi une induction mutuelle sur ces derniers. Ceci présente deux avantages par rapport à des analyses séparées ou en cascades (où chacune utiliserait le résultat des analyses précédentes). D'une part, il n'est pas nécessaire de définir quel ensemble de motifs orthogonaux doit être analysé avant quel autre. D'autre part, une induction mutuelle est strictement plus expressive. La collaboration entre les différents ensembles de motifs orthogonaux se produit lors du test de la nouvelle condition utilisée pour prouver que le membre gauche des règles raffinées n'est pas réalisable étant donné les motifs qui sont autorisés à un moment donné de l'analyse (voir en Fig. 17). Pour faire cette preuve, le raffinement d'une règle est construit de la manière habituelle. Il suffit ensuite de trouver une occurrence de protéines dans le membre gauche de la règle raffinée qui soit incompatible avec l'état actuel de l'analyse sur au moins un des ensembles de motifs orthogonaux pris en paramètre de l'analyse. Pour cela, la racine de l'ensemble de motifs orthogonaux doit être de la même sorte que l'occurrence de la protéine en question et l'information contextuelle de cette occurrence de protéines dans ce membre gauche de la règle raffinée ne doit être compatible avec aucun des motifs de cet ensemble de motifs orthogonaux déjà déjà déclarés potentiellement accessibles par l'analyse. Dans le cas contraire, l'analyseur ne peut pas prouver que le motif est inaccessible. Le motif est alors considéré comme potentiellement accessible pour la suite de l'analyse. Il s'agit bien entendu d'une approximation.

Exemple 11 *En Fig. 19, l'étape de calcul qui avait été décrite en Fig. 17 est rejouée en remplaçant le test de réalisabilité par cette procédure approchée. Au lieu de construire un plongement du membre gauche de la règle raffinée vers un complexe biochimique afin de vérifier qu'il ne contient pas de motif non encore autorisé, la nouvelle procédure se contente de vérifier pour chaque occurrence de protéines dans le membre gauche de la règle raffinée et pour chaque ensemble de motifs orthogonaux portant sur cette sorte de protéines si celui-ci contient un motif autorisé compatible avec cette occurrence. Dans ce cas, cela revient à vérifier qu'il existe bien une vue locale déjà autorisée dans laquelle les deux*

sites l et r sont liés, alors que le site c est libre et que le motif dans lequel le site r est lié et le site c est libre est autorisé dans le deuxième ensemble de motifs orthogonaux.

Outre le fait de ne pas vérifier l'existence d'un complexe biochimique qui pourrait compléter le collage obtenu entre les motifs connexes du membre gauche de la règle raffinée et les motifs déjà déclarés potentiellement accessibles par l'analyse, il est intéressant de remarquer que la procédure de décision approchée évite le calcul de tous les chevauchements entre les motifs d'intérêt non encore découverts par l'analyse, en se focalisant sur la racine de chaque ensemble de motifs orthogonaux. Ce sont les deux sources de pertes d'information dues à l'affaiblissement de la procédure de décision.

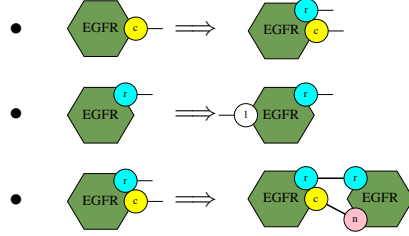
Exemple 12 *Le résultat de l'itération pour le modèle formé des règles qui avaient été décrites en Fig. 10 pour les ensembles de motifs orthogonaux qui avaient été introduits en Fig. 18, est donné en Fig. 20. Cette itération a été initialisée avec une quantité arbitraire d'occurrences de protéines de chaque sorte, mais avec tous leurs sites libres. Pour ce qui est des vues locales (voir en Fig. 20(a)), seules 4 configurations sont possibles pour l'état des sites l , r , et c des récepteurs membranaires. Ainsi, le site c ne peut être lié sans que le site r ne le soit et le site r ne peut être lié sans que le site l ne le soit. De son côté, l'analyse des doubles liaisons (voir en Fig. 20(b)) montre qu'il est impossible de former des chaînes d'au moins trois récepteurs membranaires.*

Il est important de rappeler que l'analyse ne donne qu'une sur-approximation des états accessibles. Ainsi, tout motif prouvé comme non accessible est bien inaccessible. Par contre, il n'y a aucune garantie qu'un motif non prouvé inaccessible puisse apparaître dans un état accessible depuis un des états initiaux.

5.2 Post-traitement et visualisation des résultats

L'itération de point-fixe est suivie d'une phase de traitement du résultat. Le but est essentiellement de rendre le résultat de l'analyse plus compréhensible pour l'utilisateur. Dans un premier temps, un parcours de chaque arbre de décision est effectué et chaque nœud dont tous les fils sont déclarés inaccessibles est déclaré inaccessible lui-aussi. Ensuite, tous les nœuds des arbres de décision sont explorés en répertoriant ceux dont les enfants n'ont pas tous le même statut. Ceci témoigne d'une propriété intéressante puisque dans ce cas, un des raffinements d'un motif accessible n'est pas accessible. Cette information est alors présentée sous la forme d'une implication, appelée *lemme de raffinement*, entre un motif (le nœud en question) et une liste de motifs (ses fils qui n'ont pas été prouvés inaccessibles). Une telle implication s'interprète de la manière suivante : chaque occurrence du motif de la précondition dans un complexe biochimique accessible peut toujours se raffiner en l'un des motifs de la post-condition.

Exemple 13 Le résultat de l'analyse décrit en Fig. 20 donne lieu aux implications suivantes :



Cela prouve que dans une occurrence du récepteur membranaire, le site *c* ne peut être lié sans que le site *r* ne le soit également, et que le site *r* ne peut être lié sans que le site *l* ne le soit aussi. De plus, une occurrence du récepteur dont les sites *r* et *c* sont tous deux liés, est nécessairement liée doublement à une même occurrence du récepteur.

Par ailleurs, l'analyseur vérifie pour chaque règle si son membre gauche est compatible avec le résultat de l'analyse (avec la procédure de décision simplifiée présentée Sec. 5.1). Les règles pour lesquelles ce n'est pas le cas sont reportées à l'utilisateur.

5.3 Étude de performance et utilisation concrète

Nous avons utilisé l'analyseur statique KaSa sur plusieurs modèles. Les résultats de ces analyses sont décrites en Fig. 21. Les onze premiers modèles sont des traductions directes des modèles qui sont fournis avec la distribution du logiciel BNGL (Blinov et collab. 2004). Le modèle 'sos' décrit les premiers événements de l'acquisition du facteur de croissance de l'épiderme (il comprend entre autres les règles données en Fig. 10). Les modèles 'machine' et 'ensemble' sont deux versions de la voie de signalisation MAPK, publiées par Eric Deeds et Ryan Suderman (Suderman et Deeds 2013). Les versions du modèle 'korkut' concernent la voie de signalisation de la protéine Ras. Ce modèle a été conçu par John Bachman et Benjamin Gyori (Sorgor lab) dans le cadre du projet DARPA Big Mechanism (Cohen 2015) en utilisant des outils de traitement automatique du langage naturel pour extraire des faits de la littérature, en assemblant ces faits en Kappa, puis en corrigeant manuellement le modèle obtenu (Gyori et collab. 2017). Le modèle 'tgf' s'intéresse lui à la matrice extra-cellulaire de la protéine $\text{tgf-}\beta$. Cinq versions de ce modèle ont été analysées. Elles ont été assemblées à la main d'après la littérature par Nathalie Théret et Jean Coquet, puis corrigées avec l'aide de l'analyseur KaSa. Enfin, plusieurs versions du modèle de la voie de signalisation de la protéine Wnt, écrites par Héctor F. Medina Abarca (Fontana Lab) dans le cadre du projet DARPA Big Mechanism (Cohen 2015) ont été analysées. Ce modèle a également été assemblé manuellement après lecture humaine de la littérature. Dans ce dernier modèle, le grand nombre de règles vient du fait que des scripts ont été utilisés pour raffiner des règles d'interaction génériques afin d'ajuster leur cinétique en fonction d'information

contextuelle sur l'état des occurrences de protéines qui interagissent et de leurs voisines.

Pour chaque modèle et chaque version de ce modèle, nous avons reporté le nombre total de règles dans le modèle, ainsi que nombre d'implications découvertes par l'analyse portant sur des relations soit entre au moins deux sites, soit entre les états de liaison et d'activation d'un même site. Nous avons également donné le nombre de règles qui ont été trouvées mortes par l'analyseur statique (du fait de l'approximation, l'analyseur peut manquer des règles mortes, par contre, toute règle détectée morte l'est). Nous donnons également le temps de calcul total de l'analyse, ce qui montre que KaSa passe à l'échelle même sur des modèles comportant un grand nombre de règles d'interaction.

Les informations trouvées par l'analyse statique sont utiles. Une même démarche a été suivie pour améliorer la qualité de ces modèles, qu'ils soient écrits à la main ou assemblés automatiquement par fouille automatique de la littérature. La première étape est la vérification des règles mortes. Ces règles sont souvent la conséquence, soit d'erreurs typographiques, soit d'états initiaux incomplets, soit de règles manquantes, soit de relations de causalité qui ne peuvent pas être satisfaites. La lecture des contraintes trouvées par l'analyseur permet de mieux comprendre leur origine. Elle permet également de vérifier que les invariants structurels auquel le modélisateur peut s'attendre sont bien vérifiés. L'étape suivante est d'étudier comment une occurrence de protéines peut passer d'une configuration à une autre. L'analyse des *traces locales* (Feret et Ly 2016 ; Feret et Ly 2018) calcule des systèmes de transitions à partir des vues locales. Ceci permet d'avoir une cartographie des changements de configuration de chaque occurrence de protéines en faisant abstraction de l'état des occurrences de protéines auxquelles cette occurrence est liée. En particulier, une étude de ces systèmes de transitions permet de calculer efficacement des transitions qui sont définitives : c'est à dire celles qui transforment la configuration d'une occurrence de protéines, sans retour possible, quel que soit le nombre de transitions ultérieures.

Exemple 14 *Le système de transitions qui représente les traces locales des occurrences du récepteur membranaire dans le modèles des premières interactions qui interviennent dans l'activation du facteur de croissance de l'épiderme, est dessiné en Fig. 22. La succession entre les différentes configurations possibles y est clairement décrite. Ainsi, partant d'un récepteur avec les sites l , r , c et n libres (le site n a été ajouté pour rendre l'exemple plus intéressant), le site l peut devenir lié en premier, ensuite le site r peut devenir lié, ensuite soit le site c , soit le site n peut devenir lié. Par contre, ces deux liaisons sont exclusives : les sites c et n d'une occurrence de protéines ne peuvent pas être liés tous les deux simultanément. Par ailleurs, toutes les liaisons peuvent se défaire dans l'ordre inverse de leur création.*

6 Conclusion

Après un bref passage en revue de l'écosystème Kappa et de l'utilisation de l'interprétation abstraite pour extraire les propriétés des réseaux d'interactions biomoléculaires, le langage Kappa a été présenté plus en détail, ainsi qu'une analyse statique pour détecter parmi un ensemble de motifs d'intérêt lesquels peuvent potentiellement apparaître dans des complexes biochimiques dans une trace d'exécution d'un modèle.

Du point de vue de l'utilisateur, cette analyse permet de trouver – ou de retrouver – des propriétés structurelles sur les différentes configurations des occurrences des protéines au sein des complexes biochimiques : elle détecte quelles sont les relations entre l'état des sites des occurrences d'une protéine (Est-ce que tel site peut être lié sans que tel autre le soit ? Est-ce que ce site peut être lié sans être phosphorylé ?) ; elle permet de vérifier si deux occurrences de protéines liées entre-elles sont, oui ou non, nécessairement localisées au même endroit au sein d'une hiérarchie statique de compartiments ; elle analyse si une occurrence de protéines peut être doublement liée à une autre ou si elle peut être liée à deux occurrences différentes de protéines. En plus, de permettre la détection de règles mortes, qui ne pourront jamais être appliquées dans le modèle, le résultat est présenté graphiquement sous la forme de lemmes de raffinement, ce qui le rend compréhensible et facilement utilisable pour des analyses ultérieures. Il est ensuite possible de se concentrer sur le comportement des occurrences d'une protéine en particulier et d'obtenir un système de transitions pour décrire leurs changements potentiels de configuration.

Cette analyse passe à l'échelle de grands modèles. Cependant, pour ceux-ci, le temps de calcul reste trop important pour permettre une analyse interactive et sans latence pendant l'écriture même des modèles. Une formulation du calcul du plus petit point fixe abstrait sous forme de résolution de clauses de Horn pourrait donner lieu à une analyse incrémentale. Celle-ci permettrait de mettre à jour très rapidement le résultat de l'analyse lorsque des règles sont retirées ou ajoutées à un modèle. Par ailleurs, une collaboration étroite avec les modélisateurs est toujours nécessaire pour identifier des nouvelles familles de propriétés d'intérêt. Un autre axe de recherche est l'intégration de l'analyse statique dans des cycles de modélisations automatiques. En effet, les méthodes de fouille de la littérature basées sur l'intelligence artificielle et le traitement automatique des langages naturels pourront bénéficier de l'analyse statique d'une part pour évaluer le bien fondé d'une étape de raffinement de modèle et d'autre part pour orienter les méthodes automatiques dans leur recherche de nouvelles règles.

En ce qui concerne la modélisation en Kappa, il est important de considérer non pas un réseau d'interactions biomoléculaires dans son individualité, mais une famille de réseaux d'interactions pouvant représenter un système dans différents contextes cellulaires et ses évolutions potentielles. Les travaux sur la plate-forme de modélisation Kami vont dans ce sens (Harmer et collab. 2019 ; Harmer et Oshurko 2019). Il est aussi important de proposer des méthodes pour assister le modélisateur dans la construction de modèles, afin d'agglomérer des informations partielles sur les interactions biomoléculaires en les raffinant

progressivement. Une approche inspirée des approches déductives, qui assimile le processus de modélisation à une recherche de preuves assistée par ordinateur, est très prometteuse (Husson et Krivine 2019 ; Husson 2019). Dans ce contexte, une analyse statique le plus tôt possible dans la chaîne de modélisation doit être développée pour aider au mieux le modélisateur dans sa tâche.

Améliorer l’interactivité des outils (Boutillier 2019 ; Boutillier et collab. 2018b) et un travail sur le rendu visuel des propriétés (Forbes et collab. 2018) sont des enjeux cruciaux pour créer des outils utilisables pour des modélisateurs non experts en langage formel. Il est important d’intéresser un spectre plus large d’utilisateurs. D’une part, c’est une source inépuisable de défis scientifiques qui permettent l’amélioration des outils. D’autre part, c’est nécessaire pour construire un nombre satisfaisant de modèles.

Les modèles sont de plus en plus grands, que ce soit en nombre de complexes biochimiques différents ou en nombre d’instances des complexes biochimiques. Évaluer leur comportement est primordial, mais difficile. Les méthodes exactes de réduction de modèles sont utiles, mais limitées, pour ce type de modèles. Il est important de développer des méthodes numériques approchées pour les sémantiques différentielles et stochastiques des modèles qui permettront de trouver un encadrement garanti de l’évolution du nombre d’instances ou de la concentration, selon le choix de la sémantique, de motifs d’intérêt au cours du temps, sous la forme de paires de fonctions, elles-mêmes définies comme la solution d’un système différentiel ou comme les trajectoires d’un système stochastique. Des travaux préliminaires ont permis d’intégrer dans un cadre formel des méthodes de troncation de développement formel (Saint-Germain et Feret 2016) ou des méthodes inspirées de la physique comme la tropicalisation (Beica et collab.), tout en fournissant des bornes évoluant au cours de l’exécution des modèles sur les erreurs numériques accumulées. Il devrait également être possible de définir une version quantitative de l’analyse de flot d’information entre sites des protéines, afin de négliger les petits flots d’information, au prix d’une perte de précision dans les modèles réduits. Un cadre formel pour l’exécution numériquement approchée des modèles permettra d’interfacer les sémantiques différentielles et stochastiques de Kappa pour concevoir une sémantique hybride, plus adaptée à la description des interactions entre des complexes biochimiques géants rares et des petits complexes présents en très grand nombre.

7 Bibliography

- Abou-Jaoudé, W., J. Feret et D. Thieffry. 2015, «Derivation of qualitative dynamical models from biochemical networks», dans *Computational Methods in Systems Biology - 13th International Conference, CMSB 2015, Nantes, France, September 16-18, 2015, Proceedings, Lecture Notes in Computer Science*, vol. 9308, édité par O. F. Roux et J. Bourdon, Springer, p. 195–207. URL https://doi.org/10.1007/978-3-319-23401-4_17.
- Abou-Jaoudé, W., D. Thieffry et J. Feret. 2016, «Formal derivation of qualita-

- tive dynamical models from biochemical networks», *Biosystems*, vol. 149, p. 70–112. URL <https://doi.org/10.1016/j.biosystems.2016.09.001>.
- Allart, E., J. Niehren et C. Versari. 2019, «Computing difference abstractions of metabolic networks under kinetic constraints», dans *Computational Methods in Systems Biology - 17th International Conference, CMSB 2019, Trieste, Italy, September 18-20, 2019, Proceedings, Lecture Notes in Computer Science*, vol. 11773, édité par L. Bortolussi et G. Sanguinetti, Springer, p. 266–285. URL https://doi.org/10.1007/978-3-030-31304-3_14.
- Andersen, J. L., C. Flamm, D. Merkle et P. F. Stadler. 2016, «A software package for chemically inspired graph transformation», dans *Graph Transformation - 9th International Conference, ICGT 2016, in Memory of Hartmut Ehrig, Held as Part of STAF 2016, Vienna, Austria, July 5-6, 2016, Proceedings, Lecture Notes in Computer Science*, vol. 9761, édité par R. Echahed et M. Minas, Springer, p. 73–88. URL https://doi.org/10.1007/978-3-319-40530-8_5.
- Andrei, O. et H. Kirchner. 2008, «A rewriting calculus for multigraphs with ports», *Electr. Notes Theor. Comput. Sci.*, vol. 219, p. 67–82. URL <https://doi.org/10.1016/j.entcs.2008.10.035>.
- Behr, N. et J. Krivine. 2019, «Compositionality of rewriting rules with conditions», *CoRR*, vol. abs/1904.09322. URL <http://arxiv.org/abs/1904.09322>.
- Beica, A., J. Feret et T. Petrov. «Tropical abstraction of biochemical reaction networks with guarantees», *Electr. Notes Theor. Comput. Sci.*
- Blanchet, B., P. Cousot, R. Cousot, J. Feret, L. Mauborgne, A. Miné, D. Monniaux et X. Rival. 2003, «A static analyzer for large safety-critical software», dans *Proceedings of the ACM SIGPLAN 2003 Conference on Programming Language Design and Implementation 2003, San Diego, California, USA, June 9-11, 2003*, édité par R. Cytron et R. Gupta, p. 196–207.
- Blinov, M. L., J. R. Faeder, B. Goldstein et W. S. Hlavacek. 2004, «Bionetgen: software for rule-based modeling of signal transduction based on the interactions of molecular domains», *Bioinformatics*, vol. 20, n° 17.
- Blinov, M. L., J. R. Faeder, B. Goldstein et W. S. Hlavacek. 2006, «A network model of early events in epidermal growth factor receptor signaling that accounts for combinatorial complexity», *BioSystems*, vol. 83, p. 136–151.
- Bodei, C., L. Brodo, R. Gori, D. Hermith et F. Levi. 2015, «A global occurrence counting analysis for brane calculi», dans *Logic-Based Program Synthesis and Transformation - 25th International Symposium, LOPSTR 2015, Siena, Italy, July 13-15, 2015. Revised Selected Papers, Lecture Notes in Computer Science*, vol. 9527, édité par M. Falaschi, Springer, p. 179–200. URL https://doi.org/10.1007/978-3-319-27436-2_11.
- Bodei, C., P. Degano, F. Nielson et H. R. Nielson. 1998, «Control flow analysis for the pi-calculus», dans *CONCUR '98: Concurrency Theory, 9th International Conference, Nice, France, September 8-11, 1998, Proceedings, Lecture*

- Notes in Computer Science*, vol. 1466, édité par D. Sangiorgi et R. de Simone, Springer, p. 84–98. URL <https://doi.org/10.1007/BFb0055617>.
- Borisov, N. M., N. I. Markevich, B. N. Kholodenko et E. D. Gilles. 2005, «Signaling through receptors and scaffolds: Independent interactions reduce combinatorial complexity», *Biophysical Journal*, vol. 89.
- Boutillier, P. 2019, «The kappa simulator made interactive», dans *Computational Methods in Systems Biology - 17th International Conference, CMSB 2019, Trieste, Italy, September 18-20, 2019, Proceedings, Lecture Notes in Computer Science*, vol. 11773, édité par L. Bortolussi et G. Sanguinetti, Springer, p. 296–301. URL https://doi.org/10.1007/978-3-030-31304-3_16.
- Boutillier, P., F. Camporesi, J. Coquet, J. Feret, K. Q. Lý, N. Théret et P. Vignat. 2018a, «Kasa: A static analyzer for kappa», dans *Computational Methods in Systems Biology - 16th International Conference, CMSB 2018, Brno, Czech Republic, September 12-14, 2018, Proceedings, Lecture Notes in Computer Science*, vol. 11095, édité par M. Ceska et D. Safránek, Springer, p. 285–291. URL https://doi.org/10.1007/978-3-319-99429-1_17.
- Boutillier, P., T. Ehrhard et J. Krivine. 2017, «Incremental update for graph rewriting», dans *Programming Languages and Systems - 26th European Symposium on Programming, ESOP 2017, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2017, Uppsala, Sweden, April 22-29, 2017, Proceedings, Lecture Notes in Computer Science*, vol. 10201, édité par H. Yang, Springer, p. 201–228. URL https://doi.org/10.1007/978-3-662-54434-1_8.
- Boutillier, P., A. Faure de Pebeyre et J. Feret. «Proving the absence of unbounded polymers in rule-based models», dans *Nine International Workshop on Static Analysis and Systems Biology (SASB'18)*, ENTCS, elsevier. To appear.
- Boutillier, P., M. Maasha, X. Li, H. F. Medina-Abarca, J. Krivine, J. Feret, I. Cristescu, A. G. Forbes et W. Fontana. 2018b, «The kappa platform for rule-based modeling», *Bioinformatics*, vol. 34, n° 13, p. i583–i592. URL <https://doi.org/10.1093/bioinformatics/bty272>.
- Brightman, F. A. et D. A. Fell. 2000, «Differential feedback regulation of the mapk cascade underlies the quantitative differences in egf and ngf signalling in pc12 cells», *FEBS Letters*, vol. 482, n° 3, p. 169–174.
- Camporesi, F., J. Feret, H. Koepl et T. Petrov. 2010, «Combining model reductions», dans *MFPSXXVI: Postproceedings of the 26th Conference on the Mathematical Foundations of Programming Semantics, Electronic Notes in Theoretical Computer Science*, vol. 265, Elsevier Science Publishers, p. 73–96.
- Camporesi, F., J. Feret et K. Q. Lý. 2017, «Kade: A tool to compile kappa rules into (reduced) ODE models», dans *Computational Methods in Systems Biology - 15th International Conference, CMSB 2017, Darmstadt, Germany*,

- September 27-29, 2017, *Proceedings, Lecture Notes in Computer Science*, vol. 10545, édité par J. Feret et H. Koepl, Springer, p. 291–299. URL https://doi.org/10.1007/978-3-319-67471-1_18.
- Cardelli, L. 2004, «Brane calculi», dans *Computational Methods in Systems Biology, International Conference, CMSB 2004, Paris, France, May 26-28, 2004, Revised Selected Papers, Lecture Notes in Computer Science*, vol. 3082, édité par V. Danos et V. Schächter, Springer, p. 257–278. URL https://doi.org/10.1007/978-3-540-25974-9_24.
- Cardelli, L. et A. D. Gordon. 1998, «Mobile ambients», dans *Foundations of Software Science and Computation Structure, First International Conference, FoSSaCS'98, Held as Part of the European Joint Conferences on the Theory and Practice of Software, ETAPS'98, Lisbon, Portugal, March 28 - April 4, 1998, Proceedings, Lecture Notes in Computer Science*, vol. 1378, édité par M. Nivat, Springer, p. 140–155. URL <https://doi.org/10.1007/BFb0053547>.
- Cardelli, L. et A. D. Gordon. 2000, «Mobile ambients», *Theor. Comput. Sci.*, vol. 240, n° 1, doi:10.1016/S0304-3975(99)00231-5, p. 177–213. URL [https://doi.org/10.1016/S0304-3975\(99\)00231-5](https://doi.org/10.1016/S0304-3975(99)00231-5).
- Cardelli, L., M. Tribastone, M. Tschaikowski et A. Vandin. 2015, «Forward and backward bisimulations for chemical reaction networks», dans *26th International Conference on Concurrency Theory, CONCUR 2015, Madrid, Spain, September 1-4, 2015*, p. 226–239.
- Cardelli, L., M. Tribastone, M. Tschaikowski et A. Vandin. 2019, «Symbolic computation of differential equivalences», *Theor. Comput. Sci.*, vol. 777, doi:10.1016/j.tcs.2019.03.018, p. 132–154. URL <https://doi.org/10.1016/j.tcs.2019.03.018>.
- Ciocchetta, F. et J. Hillston. 2009, «Bio-PEPA: A framework for the modelling and analysis of biological systems», *Theoretical Computer Science*, vol. 410, n° 33 – 34, p. 3065 – 3084. Concurrent Systems Biology: To Nadia Busi (1968–2007).
- Cohen, P. R. 2015, «DARPA's big mechanism program», *Physical Biology*, vol. 12, n° 4, doi:10.1088/1478-3975/12/4/045008, p. 045 008. URL <https://doi.org/10.1088/1478-3975/12/4/045008>.
- Conzelmann, H., J. Saez-Rodriguez, T. Sauter, B. N. Kholodenko et E. D. Gilles. 2006, «A domain-oriented approach to the reduction of combinatorial complexity in signal transduction networks», *BMC Bioinformatics*, vol. 7.
- Cook, B., J. Fisher, E. Krepska et N. Piterman. 2011, «Proving stabilization of biological systems», dans *Verification, Model Checking, and Abstract Interpretation - 12th International Conference, VMCAI 2011, Austin, TX, USA, January 23-25, 2011. Proceedings*, vol. 6538, édité par R. Jhala et D. A. Schmidt, Springer, p. 134–149. URL https://doi.org/10.1007/978-3-642-18275-4_11.
- Corradini, A., T. Heindel, F. Hermann et B. König. 2006, «Sesqui-pushout

- rewriting», dans *Graph Transformations, Third International Conference, ICGT 2006, Natal, Rio Grande do Norte, Brazil, September 17-23, 2006, Proceedings, Lecture Notes in Computer Science*, vol. 4178, édité par A. Corradini, H. Ehrig, U. Montanari, L. Ribeiro et G. Rozenberg, Springer, p. 30–45. URL https://doi.org/10.1007/11841883_4.
- Corradini, A., U. Montanari, F. Rossi, H. Ehrig, R. Heckel et M. Löwe. 1997, «Algebraic approaches to graph transformation - part I: basic concepts and double pushout approach», dans *Handbook of Graph Grammars and Computing by Graph Transformations, Volume 1: Foundations*, édité par G. Rozenberg, World Scientific, p. 163–246.
- Cousot, P. 1999, «The calculational design of a generic abstract interpreter», dans *Calculational System Design*, édité par M. Broy et R. Steinbrüggen, NATO ASI Series F. IOS Press, Amsterdam.
- Cousot, P. 2002, «Constructive design of a hierarchy of semantics of a transition system by abstract interpretation», *Theoretical Computer Science*, vol. 277, n° 1–2, p. 47–103.
- Cousot, P. et R. Cousot. 1977, «Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints», dans *Conference Record of the Fourth ACM Symposium on Principles of Programming Languages, Los Angeles, California, USA, January 1977*, édité par R. M. Graham, M. A. Harrison et R. Sethi, ACM, p. 238–252. URL <https://doi.org/10.1145/512950.512973>.
- Cousot, P. et R. Cousot. 1979, «Systematic design of program analysis frameworks», dans *Conference Record of the Sixth Annual ACM Symposium on Principles of Programming Languages, San Antonio, Texas, USA, January 1979*, édité par A. V. Aho, S. N. Zilles et B. K. Rosen, ACM Press, p. 269–282. URL <https://doi.org/10.1145/567752.567778>.
- Damgaard, T. C., E. Højsgaard et J. Krivine. 2012, «Formal cellular machinery», *Electr. Notes Theor. Comput. Sci.*, vol. 284, doi:10.1016/j.entcs.2012.05.015, p. 55–74. URL <https://doi.org/10.1016/j.entcs.2012.05.015>.
- Damm, W. et D. Harel. 2001, «LSCs: Breathing life into message sequence charts», *Formal Methods in System Design*, vol. 19, n° 1, doi:10.1023/A:1011227529550, p. 45–80, ISSN 1572-8102. URL <http://dx.doi.org/10.1023/A:1011227529550>.
- Danos, V., J. Feret, W. Fontana, R. Harmer, J. Hayman, J. Krivine, C. D. Thompson-Walsh et G. Winskel. 2012, «Graphs, rewriting and pathway reconstruction for rule-based models», dans *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2012, December 15-17, 2012, Hyderabad, India, LIPIcs*, vol. 18, édité par D. D’Souza, T. Kavitha et J. Radhakrishnan, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, p. 276–288. URL <https://doi.org/10.4230/LIPIcs.FSTTCS.2012.276>.
- Danos, V., J. Feret, W. Fontana, R. Harmer et J. Krivine. 2007a, «Rule-based

- modelling of cellular signalling», dans *CONCUR 2007 - Concurrency Theory, 18th International Conference, CONCUR 2007, Lisbon, Portugal, September 3-8, 2007, Proceedings, Lecture Notes in Computer Science*, vol. 4703, édité par L. Caires et V. T. Vasconcelos, Springer, p. 17–41. URL https://doi.org/10.1007/978-3-540-74407-8_3.
- Danos, V., J. Feret, W. Fontana, R. Harmer et J. Krivine. 2010, «Abstracting the differential semantics of rule-based models: Exact and automated model reduction», dans *Proceedings of the 25th Annual IEEE Symposium on Logic in Computer Science, LICS 2010, 11-14 July 2010, Edinburgh, United Kingdom*, IEEE Computer Society, ISBN 978-0-7695-4114-3, p. 362–381. URL <https://doi.org/10.1109/LICS.2010.44>.
- Danos, V., J. Feret, W. Fontana et J. Krivine. 2007b, «Scalable simulation of cellular signaling networks», dans *Programming Languages and Systems, 5th Asian Symposium, APLAS 2007, Singapore, November 29-December 1, 2007, Proceedings, Lecture Notes in Computer Science*, vol. 4807, édité par Z. Shao, Springer, p. 139–157. URL https://doi.org/10.1007/978-3-540-76637-7_10.
- Danos, V., J. Feret, W. Fontana et J. Krivine. 2008, «Abstract interpretation of cellular signalling networks», dans *Verification, Model Checking, and Abstract Interpretation, 9th International Conference, VMCAI 2008, San Francisco, USA, January 7-9, 2008, Proceedings, Lecture Notes in Computer Science*, vol. 4905, édité par F. Logozzo, D. A. Peled et L. D. Zuck, Springer, p. 83–97. URL <https://doi.org/10.1007/978-3-540-78163-9>.
- Danos, V., R. Honorato-Zimmer, S. Jaramillo-Riveri et S. Stucki. 2015, «Rigid geometric constraints for Kappa models», dans *SASB'12: PostProceedings of the 3rd International Workshop on Static Analysis and Systems Biology, ENTCS*, vol. 313, Elsevier, p. 23–46.
- Danos, V. et C. Laneve. 2003, «Graphs for core molecular biology», dans *Computational Methods in Systems Biology, First International Workshop, CMSB 2003, Roverto, Italy, February 24-26, 2003, Proceedings, Lecture Notes in Computer Science*, vol. 2602, édité par C. Priami, Springer, p. 34–46. URL https://doi.org/10.1007/3-540-36481-1_4.
- Danos, V. et C. Laneve. 2004, «Formal molecular biology», *Theoretical Computer Science*, vol. 325, n° 1, doi:<http://dx.doi.org/10.1016/j.tcs.2004.03.065>, p. 69 – 110, ISSN 0304-3975. URL <http://www.sciencedirect.com/science/article/pii/S0304397504002336>, computational Systems Biology.
- Danos, V. et S. Pradalier. 2004, «Projective brane calculus», dans *Computational Methods in Systems Biology, International Conference, CMSB 2004, Paris, France, May 26-28, 2004, Revised Selected Papers, Lecture Notes in Computer Science*, vol. 3082, édité par V. Danos et V. Schächter, Springer, p. 134–148. URL https://doi.org/10.1007/978-3-540-25974-9_11.
- Ded, T., D. Safránek, M. Troják, M. Klement, J. Salagovic et L. Brim. 2016, «Formal biochemical space with semantics in kappa and BNGL», *Electr. Notes*

- Theor. Comput. Sci.*, vol. 326, p. 27–49. URL <https://doi.org/10.1016/j.entcs.2016.09.017>.
- Dematté, L., C. Priami et A. Romanel. 2008, «The blenx language: A tutorial», dans *Formal Methods for Computational Systems Biology: 8th International School on Formal Methods for the Design of Computer, Communication, and Software Systems, SFM 2008 Bertinoro, Italy, June 2-7, 2008 Advanced Lectures*, édité par M. Bernardo, P. Degano et G. Zavattaro, Springer Berlin Heidelberg, Berlin, Heidelberg, p. 313–365. URL http://dx.doi.org/10.1007/978-3-540-68894-5_59.
- Faeder, J. R., M. L. Blinov, B. Goldstein et W. S. Hlavacek. 2005, «Rule-based modeling of biochemical networks», *Complexity*, vol. 10, n° 4, doi:10.1002/cplx.20074, p. 22–41, ISSN 1099-0526. URL <http://dx.doi.org/10.1002/cplx.20074>.
- Fähndrich, M. et F. Logozzo. 2010, «Static contract checking with abstract interpretation», dans *Formal Verification of Object-Oriented Software - International Conference, FoVeOOS 2010, Paris, France, June 28-30, 2010, Revised Selected Papers, LNCS*, vol. 6528, édité par B. Beckert et C. Marché, Springer, p. 10–30.
- Feinberg, M. «Lectures on chemical reaction networks», Notes of lectures given at the Mathematics Research Centre, University of Wisconsin, in 1979.
- Feret, J. 2000, «Confidentiality analysis of mobile systems», dans *Static Analysis, 7th International Symposium, SAS 2000, Santa Barbara, CA, USA, June 29 - July 1, 2000, Proceedings, Lecture Notes in Computer Science*, vol. 1824, édité par J. Palsberg, Springer, p. 135–154. URL https://doi.org/10.1007/978-3-540-45099-3_8.
- Feret, J. 2001, «Occurrence counting analysis for the pi-calculus», *Electr. Notes Theor. Comput. Sci.*, vol. 39, n° 2, doi:10.1016/S1571-0661(05)01155-2, p. 1–18. URL [https://doi.org/10.1016/S1571-0661\(05\)01155-2](https://doi.org/10.1016/S1571-0661(05)01155-2).
- Feret, J. 2015, «An algebraic approach for inferring and using symmetries in rule-based models», *Electr. Notes Theor. Comput. Sci.*, vol. 316, p. 45–65. URL <https://doi.org/10.1016/j.entcs.2015.06.010>.
- Feret, J., V. Danos, J. Krivine, R. Harmer et W. Fontana. 2009, «Internal coarse-graining of molecular systems», *PNAS*.
- Feret, J., H. Koepl et T. Petrov. 2013, «Stochastic fragments: A framework for the exact reduction of the stochastic semantics of rule-based models», *Int. J. Software and Informatics*, vol. 7, n° 4, p. 527–604. URL http://www.ijsi.org/ch/reader/view_abstract.aspx?file_no=i173.
- Feret, J. et K. Q. Lý. 2016, «Local traces: An over-approximation of the behaviour of the proteins in rule-based models», dans *Computational Methods in Systems Biology - 14th International Conference, CMSB 2016, Cambridge, UK, September 21-23, 2016, Proceedings, Lecture Notes in Computer Science*, vol. 9859, édité par E. Bartocci, P. Liò et N. Paoletti, Springer, p. 116–131. URL https://doi.org/10.1007/978-3-319-45177-0_8.

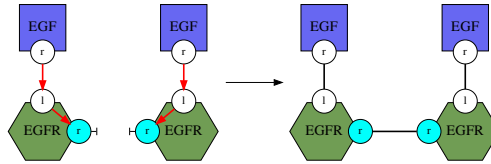
- Feret, J. et K. Q. Ly. 2018, «Local traces: An over-approximation of the behavior of the proteins in rule-based models», *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 15, n° 4, doi:10.1109/TCBB.2018.2812195, p. 1124–1137, ISSN 1545-5963. URL [doi.ieeecomputersociety.org/10.1109/TCBB.2018.2812195](https://doi.org/10.1109/TCBB.2018.2812195).
- Feret, J. et K. Q. Lý. 2018, «Reachability analysis via orthogonal sets of patterns», *Electr. Notes Theor. Comput. Sci.*, vol. 335, p. 27–48. URL <https://doi.org/10.1016/j.entcs.2018.03.007>.
- Folschette, M., L. Paulevé, M. Magnin et O. F. Roux. 2013, «Under-approximation of reachability in multivalued asynchronous networks», *Electr. Notes Theor. Comput. Sci.*, vol. 299, doi:10.1016/j.entcs.2013.11.004, p. 33–51. URL <https://doi.org/10.1016/j.entcs.2013.11.004>.
- Forbes, A. G., A. Burks, K. Lee, X. Li, P. Boutillier, J. Krivine et W. Fontana. 2018, «Dynamic influence networks for rule-based models», *IEEE Trans. Vis. Comput. Graph.*, vol. 24, n° 1, p. 184–194. URL <https://doi.org/10.1109/TVCG.2017.2745280>.
- Gao, Q., F. Liu, D. Gilbert, M. Heiner et D. Tree. 2011, «A multiscale approach to modelling planar cell polarity in drosophila wing using hierarchically coloured petri nets», dans *Proceedings of the 9th International Conference on Computational Methods in Systems Biology, CMSB '11*, ACM, New York, NY, USA, ISBN 978-1-4503-0817-5, p. 209–218, doi:10.1145/2037509.2037538. URL <http://doi.acm.org/10.1145/2037509.2037538>.
- Gay, S., F. Fages, T. Martinez, S. Soliman et C. Solnon. 2014, «On the subgraph epimorphism problem», *Discrete Applied Mathematics*, vol. 162, doi:10.1016/j.dam.2013.08.008, p. 214–228. URL <http://dx.doi.org/10.1016/j.dam.2013.08.008>.
- Gillespie, C. S. 2009, «Moment-closure approximations for mass-action models», *IET systems biology*, vol. 3, n° 1, p. 52–58.
- Gillespie, D. T. 1977, «Exact stochastic simulation of coupled chemical reactions», *The Journal of Physical Chemistry*, vol. 81, n° 25, p. 2340–2361.
- Gori, R. et F. Levi. 2006, «An analysis for proving temporal properties of biological systems», dans *Programming Languages and Systems, 4th Asian Symposium, APLAS 2006, Sydney, Australia, November 8-10, 2006, Proceedings, Lecture Notes in Computer Science*, vol. 4279, édité par N. Kobayashi, Springer, p. 234–252. URL https://doi.org/10.1007/11924661_15.
- Grosu, R., G. Batt, F. H. Fenton, J. Glimm, C. L. Guernic, S. A. Smolka et E. Bartocci. 2011, «From cardiac cells to genetic regulatory networks», dans *Computer Aided Verification - 23rd International Conference, CAV 2011, Snowbird, UT, USA, July 14-20, 2011. Proceedings, Lecture Notes in Computer Science*, vol. 6806, édité par G. Gopalakrishnan et S. Qadeer, Springer, p. 396–411. URL https://doi.org/10.1007/978-3-642-22110-1_31.
- Gyori, B., J. Bachman, K. Subramanian, J. Muhlich, L. Galescu et P. Sorger. 2017, «From word models to executable models of signaling networks using

- automated assembly», *Molecular Systems Biology*, vol. 13.
- Halpern, J. Y. et J. Pearl. 2013, «Causes and explanations: A structural-model approach — part 1: Causes», *CoRR*, vol. abs/1301.2275. URL <http://arxiv.org/abs/1301.2275>.
- Harmer, R., Y. L. Cornec, S. Légaré et E. Oshurko. 2019, «Bio-curation for cellular signalling: The KAMI project», *IEEE/ACM Trans. Comput. Biology Bioinform.*, vol. 16, n° 5, p. 1562–1573. URL <https://doi.org/10.1109/TCBB.2019.2906164>.
- Harmer, R., V. Danos, J. Feret, J. Krivine et W. Fontana. 2010, «Intrinsic information carriers in combinatorial dynamical systems», *Chaos*, vol. 20. URL <http://link.aip.org/link/CHAOEH/v20/i3/p037108/s1>.
- Harmer, R. et E. Oshurko. 2019, «Kamistudio: An environment for biocuration of cellular signalling knowledge», dans *Computational Methods in Systems Biology - 17th International Conference, CMSB 2019, Trieste, Italy, September 18-20, 2019, Proceedings, Lecture Notes in Computer Science*, vol. 11773, édité par L. Bortolussi et G. Sanguinetti, Springer, p. 322–328. URL https://doi.org/10.1007/978-3-030-31304-3_20.
- Heiner, M. et I. Koch. 2004, «Petri net based model validation in systems biology», dans *Applications and Theory of Petri Nets 2004: 25th International Conference, ICATPN 2004, Bologna, Italy, June 21–25, 2004. Proceedings*, édité par J. Cortadella et W. Reisig, Springer Berlin Heidelberg, Berlin, Heidelberg, p. 216–237. URL http://dx.doi.org/10.1007/978-3-540-27793-4_13.
- Helms, T., T. Warnke, C. Maus et A. M. Uhrmacher. 2017, «Semantics and efficient simulation algorithms of an expressive multilevel modeling language», *ACM Trans. Model. Comput. Simul.*, vol. 27, n° 2, doi:10.1145/2998499, p. 8:1–8:25. URL <https://doi.org/10.1145/2998499>.
- Husson, A. 2019, *Logical foundations of a modelling assistant for molecular biology*, thèse de doctorat, Université de Paris, France. URL http://ahusson.fr/husson_manuscript_electronic.pdf.
- Husson, A. et J. Krivine. 2019, «A tractable logic for molecular biology», dans *Proceedings 35th International Conference on Logic Programming (Technical Communications), ICLP 2019 Technical Communications, Las Cruces, NM, USA, September 20-25, 2019., EPTCS*, vol. 306, édité par B. Bogaerts, E. Erdem, P. Fodor, A. Formisano, G. Ianni, D. Incezan, G. Vidal, A. Villanueva, M. D. Vos et F. Yang, p. 101–113. URL <https://doi.org/10.4204/EPTCS.306.17>.
- John, M., C. Lhoussaine, J. Niehren et C. Versari. 2011, «Biochemical reaction rules with constraints», dans *Programming Languages and Systems - 20th European Symposium on Programming, ESOP 2011, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2011, Saarbrücken, Germany, March 26-April 3, 2011. Proceedings, Lecture Notes in Computer Science*, vol. 6602, Springer, p. 338–357. URL http://dx.doi.org/10.1007/978-3-642-19935-9_24.

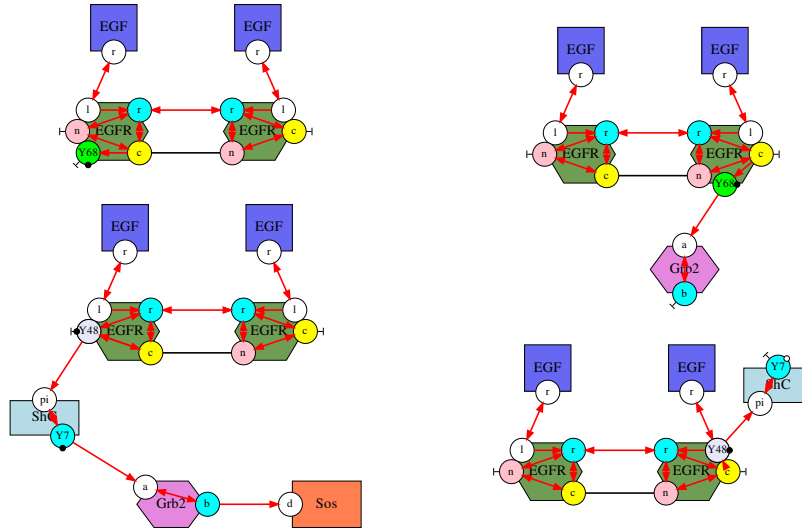
- org/10.1007/978-3-642-19718-5.
- John, M., M. Nebut et J. Niehren. 2013, «Knockout prediction for reaction networks with partial kinetic information», dans *Verification, Model Checking, and Abstract Interpretation, 14th International Conference, VMCAI 2013, Rome, Italy, January 20-22, 2013. Proceedings, Lecture Notes in Computer Science*, vol. 7737, édité par R. Giacobazzi, J. Berdine et I. Mastroeni, Springer, p. 355–374. URL https://doi.org/10.1007/978-3-642-35873-9_22.
- Kahramanogullari, O. et L. Cardelli. 2013, «An intuitive modelling interface for systems biology», *Int. J. Software and Informatics*, vol. 7, n° 4, p. 655–674. URL http://www.ijsi.org/ch/reader/view_abstract.aspx?file_no=i177.
- Klarner, H., A. Bockmayr et H. Siebert. 2015, «Computing maximal and minimal trap spaces of boolean networks», *Natural Computing*, vol. 14, n° 4, doi: 10.1007/s11047-015-9520-7, p. 535–544. URL <https://doi.org/10.1007/s11047-015-9520-7>.
- Köhler, A., J. Krivine et J. Vidmar. 2014, «A rule-based model of base excision repair», dans *Computational Methods in Systems Biology - 12th International Conference, CMSB 2014, Manchester, UK, November 17-19, 2014, Proceedings, Lecture Notes in Computer Science*, vol. 8859, édité par P. Mendes, J. O. Dada et K. Smallbone, Springer, p. 173–195. URL https://doi.org/10.1007/978-3-319-12982-2_13.
- Kolcák, J., D. Safránek, S. Haar et L. Paulevé. 2019, «Parameter space abstraction and unfolding semantics of discrete regulatory networks», *Theor. Comput. Sci.*, vol. 765, doi:10.1016/j.tcs.2018.03.009, p. 120–144. URL <https://doi.org/10.1016/j.tcs.2018.03.009>.
- Kwiatkowska, M. Z., G. Norman et D. Parker. 2011, «PRISM 4.0: Verification of probabilistic real-time systems», dans *Computer Aided Verification - 23rd International Conference, CAV 2011, Snowbird, UT, USA, July 14-20, 2011. Proceedings, Lecture Notes in Computer Science*, vol. 6806, édité par G. Gopalakrishnan et S. Qadeer, Springer, p. 585–591. URL https://doi.org/10.1007/978-3-642-22110-1_47.
- Laurent, J., J. Yang et W. Fontana. 2018, «Counterfactual resimulation for causal analysis of rule-based models», dans *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden.*, édité par J. Lang, ijcai.org, p. 1882–1890. URL <https://doi.org/10.24963/ijcai.2018/260>.
- Löwe, M. 1993, «Algebraic approach to single-pushout graph transformation», *Theor. Comput. Sci.*, vol. 109, n° 1&2, p. 181–224. URL [https://doi.org/10.1016/0304-3975\(93\)90068-5](https://doi.org/10.1016/0304-3975(93)90068-5).
- Mazurkiewicz, A. W. 1984, «Traces, histories, graphs: Instances of a process monoid», dans *Mathematical Foundations of Computer Science 1984, Praha, Czechoslovakia, September 3-7, 1984, Proceedings, Lecture Notes in Computer*

- Science*, vol. 176, édité par M. Chytil et V. Koubek, Springer, p. 115–133. URL <https://doi.org/10.1007/BFb0030293>.
- Nielson, H. R. et F. Nielson. 2000, «Shape analysis for mobile ambients», dans *POPL 2000, Proceedings of the 27th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, Boston, Massachusetts, USA, January 19-21, 2000*, édité par M. N. Wegman et T. W. Reps, ACM, p. 142–154. URL <https://doi.org/10.1145/325694.325711>.
- Paulevé, L., M. Magnin et O. F. Roux. 2011, «Abstract interpretation of dynamics of biological regulatory networks», *Electr. Notes Theor. Comput. Sci.*, vol. 272, p. 43–56. URL <https://doi.org/10.1016/j.entcs.2011.04.004>.
- Petrov, T., J. Feret et H. Koepl. 2012, «Reconstructing species-based dynamics from reduced stochastic rule-based models», dans *Winter Simulation Conference, WSC '12, Berlin, Germany, December 9-12, 2012*, édité par O. Rose et A. M. Uhrmacher, WSC, ISBN 978-1-4673-4779-2, p. 225:1–225:15. URL <https://doi.org/10.1109/WSC.2012.6465241>.
- Plateau, B. 1985, «On the stochastic structure of parallelism and synchronization models for distributed algorithms», *SIGMETRICS Perform. Eval. Rev.*, vol. 13, n° 2, doi:10.1145/317786.317819, p. 147–154, ISSN 0163-5999. URL <http://doi.acm.org/10.1145/317786.317819>.
- Radulescu, O., A. N. Gorban, A. Zinovyev et V. Noel. 2012, «Reduction of dynamical biochemical reactions networks in computational biology», *Frontiers in Genetics*, vol. 3, p. 131.
- Radulescu, O., S. Vakulenko et D. Grigoriev. 2015, «Model reduction of biochemical reactions networks by tropical analysis methods», *Mathematical Modelling of Natural Phenomena*, vol. 10, n° 3, p. 124–138.
- Regev, A., E. M. Panina, W. Silverman, L. Cardelli et E. Y. Shapiro. 2004, «Bioambients: An abstraction for biological compartments», *TCS*, vol. 325, n° 1, p. 141–167.
- Regev, A., W. Silverman et E. Shapiro. 2001, «Representation and simulation of biochemical processes using the pi-calculus process algebra», dans *Pacific Symposium on Biocomputing, Volume 6*, édité par R. B. Altman, A. K. Dunker, L. Hunter et T. E. Klein, Singapore, p. 459–470.
- Saint-Germain, K. C. et J. Feret. 2016, «Conservative numerical approximations of the differential semantics in biological rule-based models», Master thesis.
- Schoeberl, B., C. Eichler-Jonsson, E. D. Gilles et G. Müller. 2002, «Computational modeling of the dynamics of the map kinase cascade activated by surface and internalized egf receptors.», *Nat Biotechnol*, vol. 20, n° 4, p. 370–375.
- Stewart, D. 2010, «Spatial biomodelling», Master thesis, School of Informatics, University of Edinburgh.
- Suderman, R. et E. J. Deeds. 2013, «Machines vs. ensembles: effective mapk signaling through heterogeneous sets of protein complexes.», *PLoS Computational Biology*, vol. 9.

Tarski, A. 1955, «A lattice-theoretical fixpoint theorem and its applications.»,
Pacific J. Math., vol. 5, n° 2.



(a) La règle de formation de dimers annotée par le flot d'information qu'elle induit.



(b) Les quatre motifs d'intérêt qui apparaissent dans le complexe biochimique de la Fig. 1(a).

Figure 4: En 4(a), chaque chemin entre un site dont l'état est testé et un site dont l'état est modifié dans une composante connexe du membre gauche d'une règle induit un flot d'information. Ici, la capacité de lier le site r d'une occurrence du récepteur dépend du fait que cette occurrence soit liée à une occurrence du ligand. En 4(b) sont représentés les quatre motifs d'intérêt qui apparaissent dans le complexe biochimique dessiné en Fig. 1(a). Ils sont tous quatre annotés par une relation qui spécifie comment l'information se propage – ou s'est propagée – à travers leurs différents sites d'interaction (cette relation est obtenue en recopiant le flot d'information des règles compatibles avec ces motifs). Ils contiennent chacun un site accessible par tous les autres en suivant cette relation.

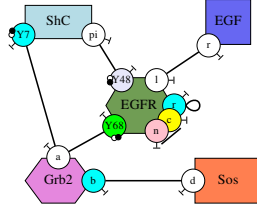


Figure 5: Une carte de contacts. Elle définit la signature d'un modèle en donnant la liste de toutes les sortes de protéines, leurs différents sites d'interaction, les différents états internes que peuvent prendre ces sites et les différentes liaisons potentielles entre ces sites.

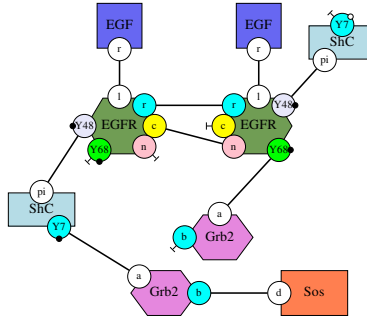


Figure 6: Un complexe biochimique. Il contient plusieurs occurrences de protéines. Chaque occurrence documente l'ensemble de ses sites d'interaction. Les sites qui peuvent porter un état interne en ont un. Par ailleurs, les sites sont soit libres, soit liés deux à deux.

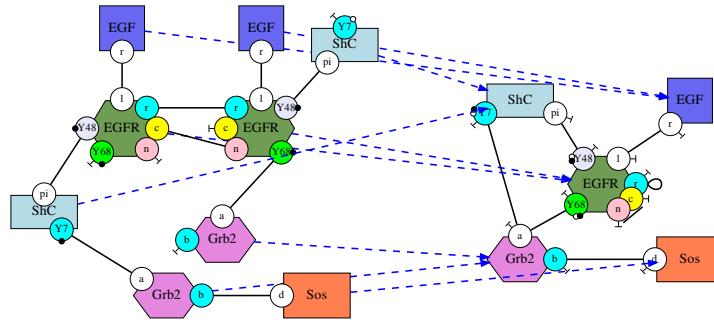


Figure 7: L'unique projection entre le complexe biochimique de la Fig. 6 et la carte de contacts de la Fig. 5. Cette projection est obtenue en associant chaque occurrence de protéines de l'espèce biochimique à l'unique sorte de protéines correspondante dans la carte de contacts.

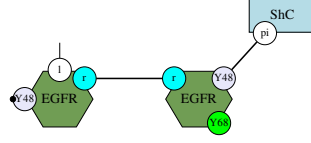


Figure 8: Un motif connexe. Il contient plusieurs occurrences de protéines. Chaque occurrence de protéines documente un sous ensemble de ses sites d'interaction. Chaque site peut éventuellement porter un état interne et éventuellement un état de liaison (en conformité avec la signature du modèle, donnée en Fig. 5). Comme état de liaison, un site peut être libre, lié sans que le site partenaire ne soit précisé ou être lié à un autre site.

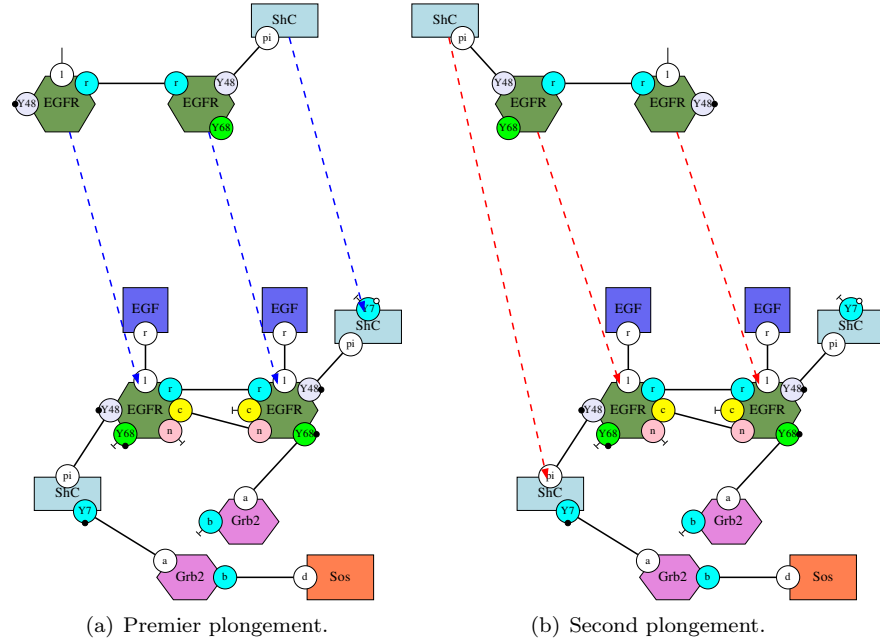


Figure 9: Deux plongements entre le motif donné dans la Fig. 8 et le complexe biochimique donné dans la Fig. 6. En 9(a) l'occurrence de la protéine d'échafaudage est associée à l'occurrence de la protéine d'échafaudage dont le site Y7 est libre. En 9(b) l'occurrence de la protéine d'échafaudage est associée à l'occurrence de la protéine d'échafaudage dont le site Y7 est lié.

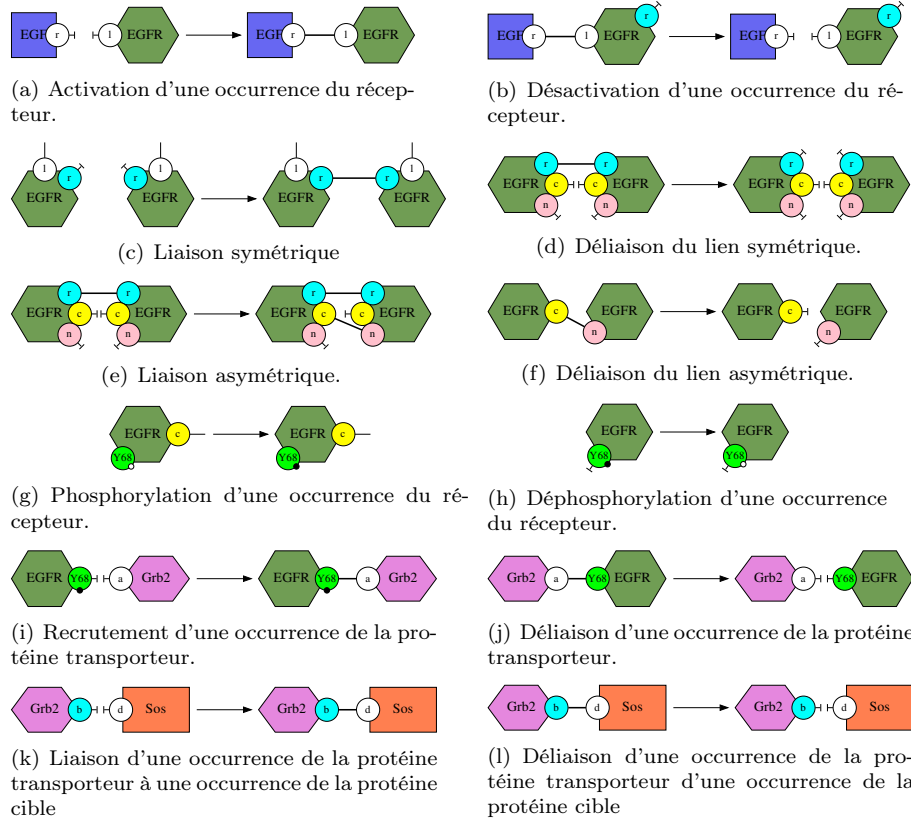
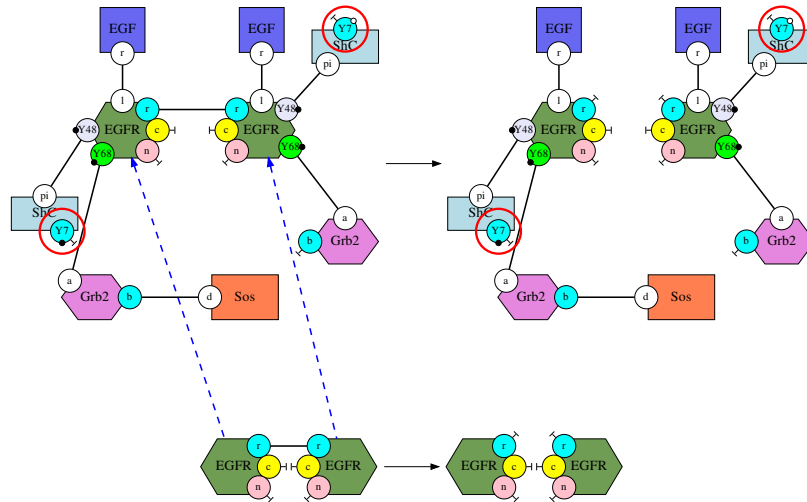
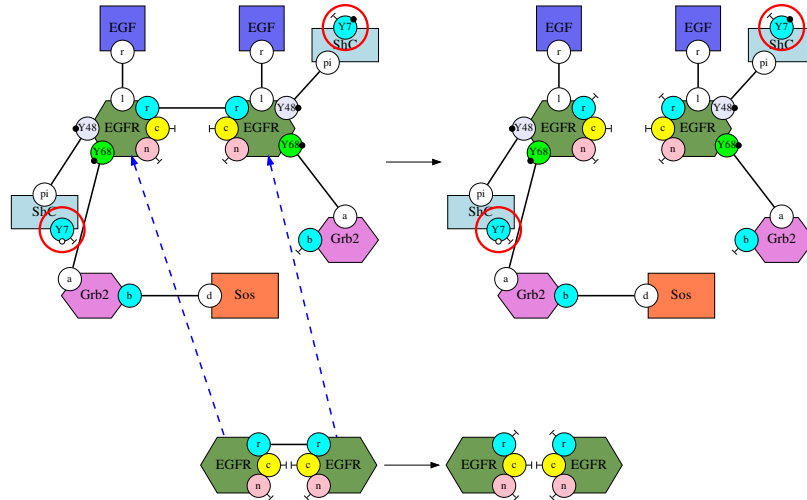


Figure 10: Règles d'interaction impliquées dans le recrutement d'une occurrence de la protéine cible par la voie de signalisation courte (sans passer par la protéine d'échafaudage).



(a) Premier raffinement.



(b) Second raffinement.

Figure 11: Deux exemples de raffinements d'une même règle d'interaction en deux règles-réactions. Les différences entre ces deux raffinements sont mises en valeur par des cercles rouges (les deux occurrences de la protéine *Shc* ont été échangées). Dans les deux cas, la règle-réaction est obtenue en ajoutant dans le membre gauche et dans le membre droit de la règle d'interaction exactement la même information sur le contexte d'application de la règle.

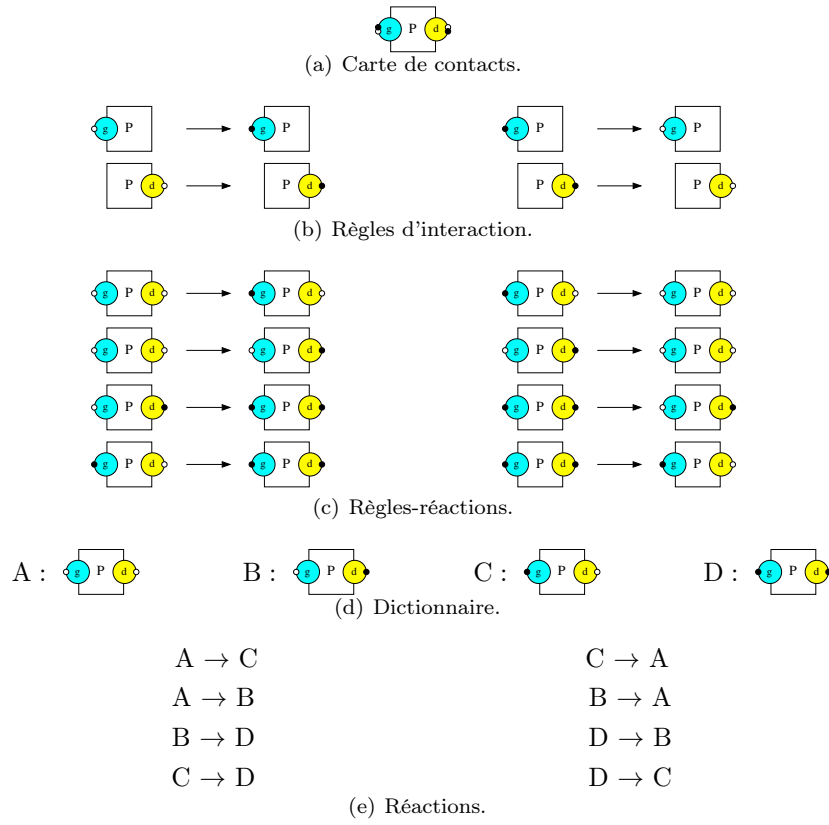
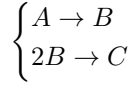
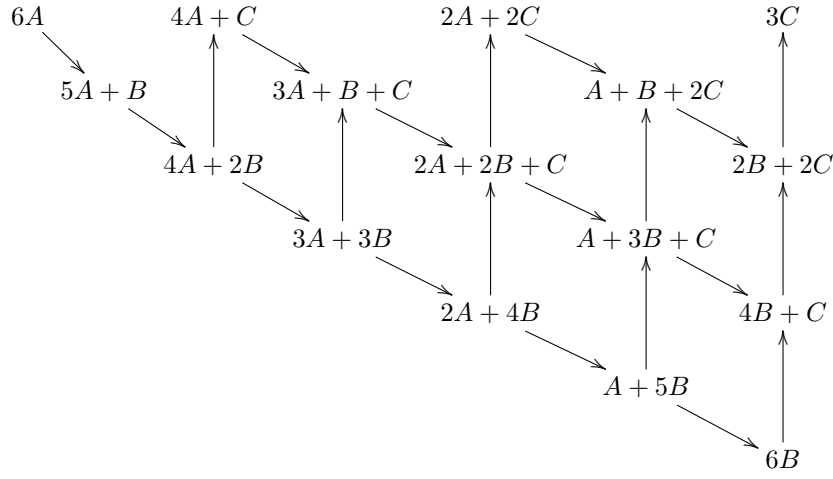


Figure 12: Un modèle formé d'une carte de contacts et de quatre règles d'interaction et sa traduction sous forme réseau réactionnel.



(a) Un réseau réactionnel.



(b) Système de transitions sous-jacent.

Figure 13: Un réseau réactionnel et sa sémantique. En 13(a) un réseau réactionnel formé de deux réactions. La première permet de transformer une occurrence de l'espèce biochimique A en une occurrence de l'espèce biochimique B , la seconde permet de transformer deux occurrences de l'espèce biochimique B en une occurrence de l'espèce biochimique C . La restriction de l'ensemble de toutes les transitions possibles aux états qui sont atteignables à partir d'un état initial formé de six occurrences de la protéine A est dessinée en 13(b) sous la forme d'un système de transitions.

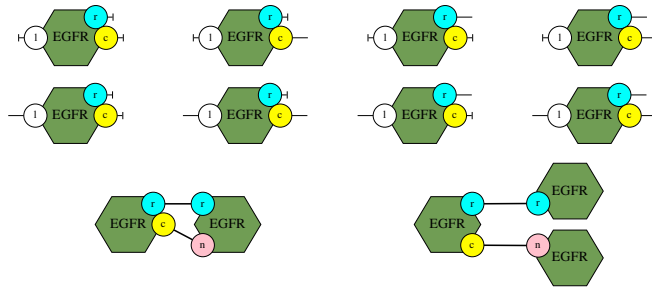
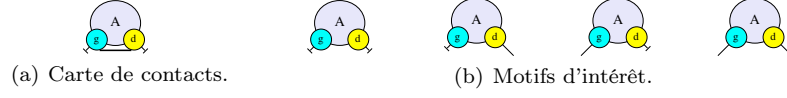


Figure 14: Un ensemble de motifs d'intérêt pour l'analyse des complexes biochimiques accessibles dans le modèle des premières interactions qui interviennent dans l'acquisition du facteur de croissance de l'épiderme. Les huit premiers motifs permettent de s'intéresser aux relations potentielles entre l'état de liaison des sites l , r et c dans les occurrences du récepteur membranaire. Ces 8 motifs correspondent exactement à chaque combinaison possible pour l'état de ces 3 sites, chacun de ces sites pouvant être libre ou lié. Les deux derniers motifs permettent de distinguer deux occurrences du récepteur liées par une double liaison d'une chaîne d'au moins trois occurrences du récepteur.



- $\alpha_{\mathcal{P}} \left(\left\{ \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \right\} \right) = \left\{ \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \right\}$
 - $\alpha_{\mathcal{P}} \left(\left\{ \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array}, \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \right\} \right) = \left\{ \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array}, \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \right\}$
 - $\alpha_{\mathcal{P}} \left(\left\{ \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array}, \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array}, \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \right\} \right) = \left\{ \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array}, \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array}, \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \right\}$
 - $\alpha_{\mathcal{P}} \left(\left\{ \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \right\} \right) = \left\{ \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \right\}$
 - $\alpha_{\mathcal{P}} \left(\left\{ \begin{array}{c} \text{A} \text{---} \text{s} \text{---} \text{d} \text{---} \text{A} \\ \text{A} \text{---} \text{s} \text{---} \text{d} \end{array} \right\} \right) = \left\{ \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \right\}$
- (c) Exemples de meilleures approximations.

- $\gamma_{\mathcal{P}} \left(\left\{ \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \right\} \right) = \left\{ n \cdot \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \mid n \in \mathbb{N} \right\}$
- $\gamma_{\mathcal{P}} \left(\left\{ \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \right\} \right) = \left\{ n_1 \cdot \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} + n_2 \cdot \begin{array}{c} \text{A} \text{---} \text{s} \text{---} \text{d} \text{---} \text{A} \\ \text{A} \text{---} \text{s} \text{---} \text{d} \end{array} + \dots \mid n_1, n_2, \dots \in \mathbb{N} \right\}$
- $\gamma_{\mathcal{P}} \left(\left\{ \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \right\} \right) = \{\emptyset\}.$
- $\gamma_{\mathcal{P}} \left(\left\{ \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \right\} \right) = \{\emptyset\}.$
- $\gamma_{\mathcal{P}} \left(\left\{ \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array}, \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \right\} \right) = \left\{ n \cdot \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array}, \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \mid n \in \mathbb{N} \right\}.$
- $\gamma_{\mathcal{P}} \left(\left(\left\{ \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array}, \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array}, \begin{array}{c} \text{A} \\ \text{s} \text{---} \text{d} \end{array} \right\} \right) \right) = \left\{ \begin{array}{l} n_2 \cdot \begin{array}{c} \text{A} \text{---} \text{s} \text{---} \text{d} \text{---} \text{A} \\ \text{A} \text{---} \text{s} \text{---} \text{d} \end{array} \\ + n_3 \cdot \begin{array}{c} \text{A} \text{---} \text{s} \text{---} \text{d} \text{---} \text{A} \text{---} \text{s} \text{---} \text{d} \text{---} \text{A} \\ \text{A} \text{---} \text{s} \text{---} \text{d} \end{array} + \dots \\ + n'_1 \cdot \begin{array}{c} \text{A} \text{---} \text{s} \text{---} \text{d} \end{array} + n'_2 \cdot \begin{array}{c} \text{A} \text{---} \text{s} \text{---} \text{d} \end{array} + \dots \end{array} \mid \begin{array}{l} n_2, n_3, \dots \in \mathbb{N} \\ n'_1, n'_2, \dots \in \mathbb{N} \end{array} \right\}.$

(d) Exemples de concrétisations

Figure 15: Un exemple jouet pour mieux comprendre le comportement des fonctions d'abstraction et de concrétisation. En 15(a), la signature du modèle : une seule sorte de protéines, A , avec deux sites pouvant être libres ou liés à l'autre site de la même ou d'une autre occurrence de la protéine A . En 15(b), le domaine abstrait est formé des vues locales de l'unique sorte de protéines : toutes les configurations pour les occurrences de la protéine A sont considérées selon que chaque site soit libre ou lié. En 15(c) sont donnés des exemples de meilleure approximation d'ensemble d'états. Cela consiste à collecter les vues locales qui peuvent apparaître dans ces états. Réciproquement, En 15(d) sont donnés des exemples de concrétisations d'ensembles de vues locales. Ceci consiste à recomposer l'ensemble des états qui ne contiennent aucune occurrence des vues locales manquantes. Dans le cas particulier des vues locales, cela revient à prendre en compte tous les états composés uniquement des vues locales mises à disposition, sachant que chaque vue peut être utilisée zéro, une ou plusieurs fois.

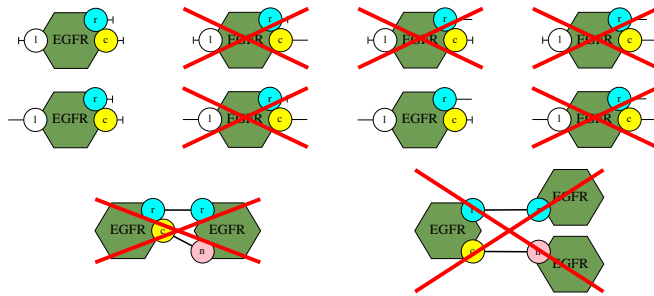
$$\gamma_{\mathcal{P}} \left(\alpha_{\mathcal{P}} \left(\text{Diagram 1} \right) \right) = \left\{ n_1 \cdot \text{Diagram 2} + n_2 \cdot \text{Diagram 3} + \dots \mid n_1, n_2, \dots \in \mathbb{N} \right\}$$

(a) Exemple d'application de la fonction $\gamma_{\mathcal{P}} \circ \alpha_{\mathcal{P}}$.

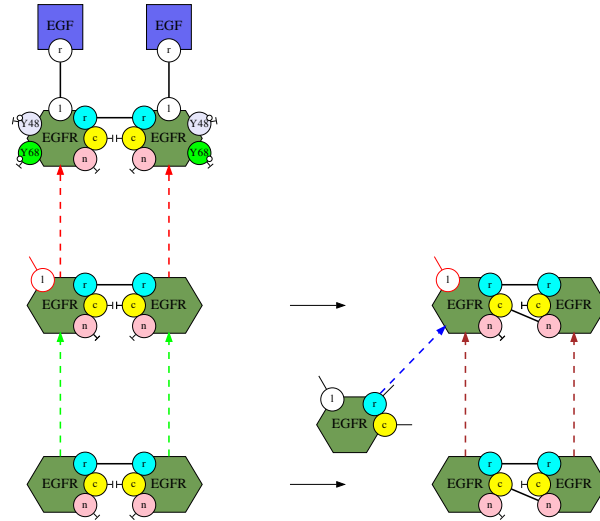
$$\alpha_{\mathcal{P}} \left(\gamma_{\mathcal{P}} \left(\text{Diagram 4}, \text{Diagram 5} \right) \right) = \left\{ \text{Diagram 6} \right\}$$

(b) Exemple d'application de la fonction $\alpha_{\mathcal{P}} \circ \gamma_{\mathcal{P}}$.

Figure 16: Suite de l'exemple donné en Fig. 15. Un exemple d'application de la composée de fonctions $\gamma_{\mathcal{P}} \circ \alpha_{\mathcal{P}}$ est montré en 16(a). Celui-ci montre que l'abstraction ne permet pas de distinguer des ensembles d'anneaux d'occurrences de la protéine A et ce quels que soient leurs tailles et leurs nombres. En 16(b) donne un exemple d'application de la composée de fonctions $\alpha_{\mathcal{P}} \circ \gamma_{\mathcal{P}}$. Cette fonction calcule que la vue locale avec le site g libre et le site d lié ne peut pas apparaître dans une espèce biochimique qui ne contiendrait pas la vue avec le site g lié et le site d libre.

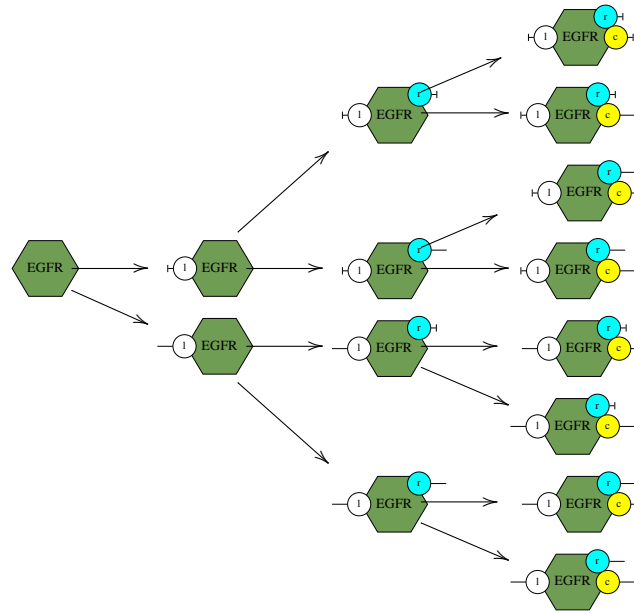


(a) Un état abstrait.

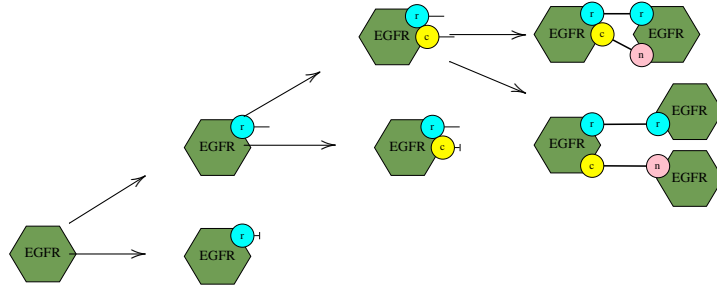


(b) La construction de la vue locale dans laquelle les trois sites sont liés peut être construite en une étape à partir de cet état abstrait.

Figure 17: Découverte d'un nouveau motif d'intérêt accessible dans le modèle des premières étapes de la voie de l'acquisition du facteur de croissance de d'épiderme (voir en Fig. 10 page 54). En 17(a), il est supposé qu'à ce moment de l'analyse, seules trois vues locales sont autorisées. Par ailleurs, il n'est permis de former ni des doubles liaisons entre récepteurs membranaires, ni des chaînes de trois récepteurs ou plus. En 17(b), la preuve que la vue locale doit être déclarée accessible à ce niveau d'abstraction est représentée sous forme de diagramme. Elle consiste à appliquer la règle de liaison asymétrique en identifiant la vue locale au récepteur de gauche dans le membre droit de la règle et en raffinant la règle en conséquence. Le membre gauche de la règle obtenue apparaît dans une espèce biochimique ne contenant aucun motif d'intérêt non encore découvert, ce qui conclut la preuve.



(a) Ensemble de motifs orthogonaux pour les vues locales.



(b) Ensemble de motifs orthogonaux pour discuter des doubles liaisons.

Figure 18: Deux exemples d'ensemble de motifs orthogonaux. En 18(a), l'ensemble des vues locales (voir les huit premiers motifs en Fig. 14 page 58) sous forme d'arbre de décision. En 18(b), celui pour discuter de la présence potentielle de doubles liaisons entre des récepteurs et de la présence potentielle de trimers (voir les deux derniers motifs en Fig. 14).

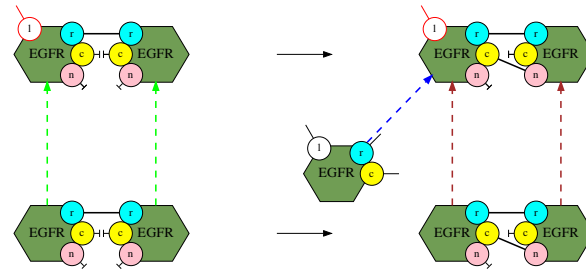
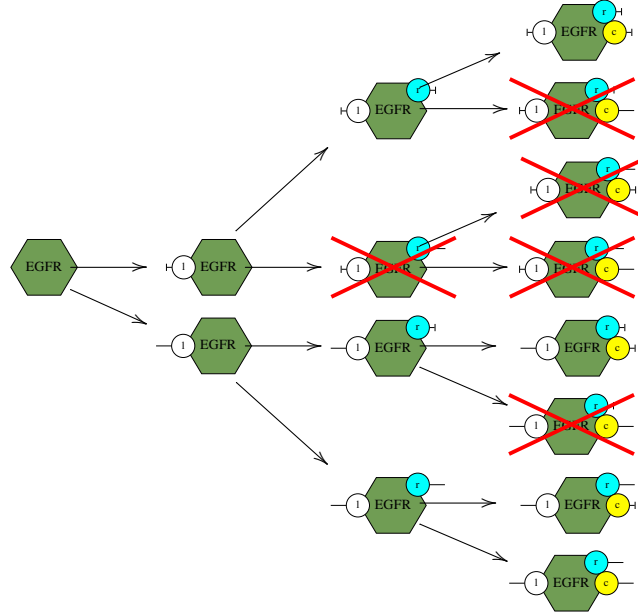
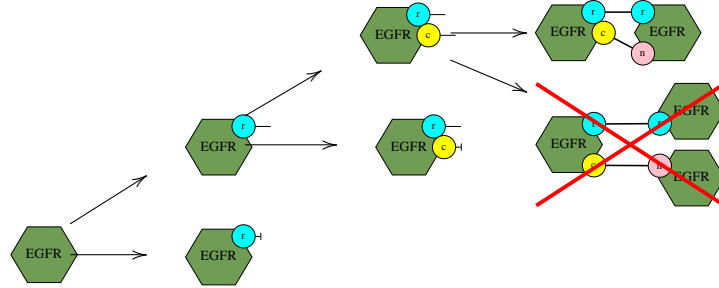


Figure 19: L'étape abstraite de la Fig. 17 revisitée avec la procédure de décision approchée. Au lieu de vérifier que chaque motif connexe du membre gauche de la règle raffinée se plonge dans un complexe biochimique dans la concrétisation de l'état abstrait, il suffit de s'assurer, pour chaque occurrence de protéines dans ce motif et chaque ensemble de motifs orthogonaux portant sur ce type de protéines si il contient un motif compatible déjà découvert par l'analyse.



(a) Analyse d'accessibilité pour l'ensemble des vues locales (voir en Fig. 18(a)).



(b) Analyse d'accessibilité pour discuter de la présence éventuelle de trimers et de double liaisons.

Figure 20: Résultat de l'analyse pour les deux ensembles de motifs orthogonaux donnés en Fig. 18. Les motifs orthogonaux sont aux feuilles des arbres de décision. Ceux qui sont barrés en rouge n'apparaissent dans aucune exécution du modèle (pour n'importe quel état initial sans lien). Par construction de l'arbre de décision, les nœuds dont tous les enfants sont inaccessibles sont également inaccessibles et donc barrés eux-aussi.

modèle	nombre de règles	nombre de contraintes inférées	nombre de règles mortes détectées	temps d'analyse (secondes)
repressilator	42	0	0	0.010
egfr_net	39	4	0	0.024
egfr_net_red	45	6	0	0.027
fceri_fyn	46	4	0	0.054
fceri_fyn_lig	48	4	0	0.054
fceri_fyn_trimer	362	4	36	0.649
fceri_fyn_gamma2	59	5	0	0.075
fceri_fyn_ji	36	4	0	0.040
fceri_fyn_ji_red	32	4	0	0.036
fceri_fyn_lyn_745	40	4	2	0.046
fceri_fyn_trimer	192	4	0	0.329
sos	20	9	0	0.023
machine	220	13	7	0.904
ensemble	233	26	0	0.804
korkut (2017/01/13)	3916	0	1610	17
korkut (2017/01/17)	12896	0	874	53
korkut (2017/02/06)	5750	0	884	114
TGF (V19)	97	19	10	0.459
TGF (V20)	99	30	10	0.563
TGF (V21)	211	18	0	1.37
TGF (2017/04/01)	235	13	0	1.09
TGF (2018/04/19)	292	13	0	1.24
BigWnt (2015/12/28)	356	2	1	4.67
BigWnt (2016/09/28)	1419	1	0	14
BigWnt (2017/03/22)	1486	14	12	19

Figure 21: Résultats expérimentaux (calculés sur un MacBook Pro avec une puce Intel Core i7-6567U (cadencée 3.3 GHz)). Pour chaque modèle et chaque version, le nombre de règles est donné, ainsi que le nombre de contraintes découvertes par l'analyse et le nombre de règles mortes trouvées (qui ne sont donc jamais utilisées dans le modèle). Le temps total de l'analyse est également fourni.

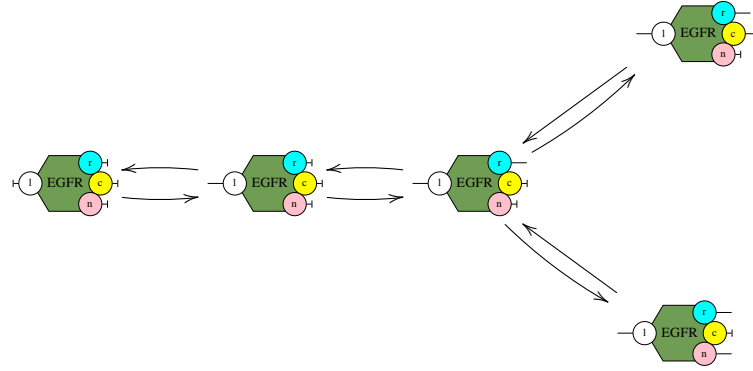


Figure 22: Le système de transitions pour les vues locales des occurrences du récepteur membranaire. L'état du site n a été ajouté pour rendre l'exemple plus intéressant. Ce système de transitions explique les étapes que traverse ces occurrences lorsque leurs sites deviennent liés. Les transitions en sens inverse, qui correspondent aux règles de libération des sites sont aussi représentées.

Index

état (réseau réactionnel), 24
état d'activation, 15
état de liaison, 15

carte de contacts, 14
chevauchements (de motifs), 30
complexe biochimique, 16
concrétisation, 26
concrétisation (fonction de), 26
contre-partie abstraite, 29
correspondance de Galois, 27

ensemble de motifs orthogonaux, 31

homomorphisme, 17

lemme de raffinement, 34

meilleure approximation, 26
motif, 17

plongement, 18

règle d'interaction, 20
règle morte, 24
règle-réaction, 21
rigidité, 19

site d'interaction, 15
sorte de protéines, 14

théorème de Tarski, 25
trace locale, 35
transition (réseau réactionnel), 24

vue locale, 26